



Trabajo Fin de Grado

Plataforma web para compartir información sobre
el uso de las tecnologías de Big Data

Web platform to sharing information on the use of
Big Data technologies

Autor

Ian Paul Ávila Matos

Director

Jaime González Pina

PONENTE

Francisco Javier López Pellicer

EINA/ Ingeniería Informática

2020

Resumen ejecutivo

TECNARA es el Clúster de empresas TIC, Electrónica y Telecomunicaciones de Aragón, la cual aglutina a las empresas del sector en la comunidad. Está reconocido como agrupación empresarial innovadora por el Ministerio de Industria, Energía y Turismo siendo su principal objetivo el de ayudar en la estrategia y desarrollo de negocio de sus socios. En diciembre de 2019 el Instituto Aragonés de Empleo le adjudicó un proyecto experimental para ver nuevas formas de captación y formación de perfiles tecnológicos junto a las empresas del sector, buscando con esta formación cubrir diferentes ámbitos dentro del sector, entre ellos el de la analítica de grandes volúmenes de datos.

Problema

El mayor desafío de TECNARA se centra en la recopilación de información, debido a que existe una ausencia de información que muestre las herramientas de Big Data más utilizadas, las ventajas de cada una de ellas o las necesidades que estas cubren, así como tener recopilado, además, el conjunto de necesidades y experiencias de cada empresa en lo referente al manejo de herramientas de datos, para conseguir un mayor aprovechamiento de los cursos ofertados.

Solución

La solución es una plataforma web denominada “TECNARA Tools”, la cual pueda ser utilizada para obtener información del conjunto de empresas de IT de Aragón independientemente del dominio estudiado. Además, se adaptará este sistema genérico al dominio específico requerido, el Big Data, creando un demostrador. Este demostrador, permitirá a las empresas compartir información sobre las herramientas de Big Data, generando una serie de informes agregados que faciliten la toma de decisiones.

Índice

<i>Resumen ejecutivo</i>	<i>ii</i>
<i>Índice</i>	<i>iii</i>
<i>Lista de figuras</i>	<i>v</i>
1. Introducción	1
1.1. Aragón y el Big Data	1
1.2. Problemas abordados	4
1.3. Objetivos del proyecto	4
1.4. Organización y gestión del proyecto	5
1.5. Estructura de la memoria	6
2. TECNARA Tools	7
2.1. Requisitos iniciales	7
2.2. Análisis del sistema	8
2.3. Usuarios del sistema	9
2.4. Casos de uso	9
2.5. Requisitos identificados	11
2.6. Diseño del sistema	12
2.7. Implementación	14
2.8. Pruebas	15
3. TECNARA Tools aplicado a un caso: Big Data	17
3.1. Caso de aplicación	17
3.2. Estrategia	17
3.3. Creación de contenidos	17
3.3.1 Definición de la pregunta de investigación y sus objetivos	17
3.3.2 Metodología	17
3.3.3 Resultados	19
3.4. Diseño del demostrador	21
3.5. Despliegue y carga de datos	21
4. Conclusiones	23
4.1. Sobre TECNARA Tools	23
4.2. Sobre la construcción del demostrador de Big Data con TECNARA tools	23
4.3. Sobre mi aprendizaje	24
4.4. Sobre el futuro de TECNARA Tools	24
Bibliografía	25
A Diccionario de términos	26
A.1 Acrónimos	26
A.2 Diccionario	26
B Gestión del proyecto	29
C Análisis sistemático	32
C.1 Contraste con la competencia	33
D Historias de usuario	34
E Requisitos de la aplicación	36
F Tecnologías utilizadas para la elaboración del portal	38
G Diagramas de Caso de Uso	39
H Vista de módulos	45

<i>I</i>	<i>Diagrama de despliegue</i>	48
<i>J</i>	<i>Modelado de datos</i>	50
J.1	Sincronización entre ElasticSearch & BBDD	52
J.2	Búsquedas en ElasticSearch	54
J.3	Protección de rutas	55
J.4	Diferencias entre roles	55
J.5	Descargar Informes	56
J.6	Gráficas	58
J.7	Seguridad	58
J.8	Diagramas de secuencia	59
<i>K</i>	<i>Mapa de navegación</i>	68
<i>L</i>	<i>Interfaces de usuario</i>	71
<i>M</i>	<i>Guía de Despliegue</i>	79

Lista de figuras

Figura 1 Ranking de actividades económicas de Aragón	1
Figura 2 Análisis de las características que delimitan Big Data	2
Figura 3 Casos de uso: Usuario no registrado.....	9
Figura 4 Casos de uso: Usuario registrado.....	10
Figura 5 Casos de uso: Administrador.....	11
Figura 6 Diagrama arquitectural lógico	12
Figura 7 Vista de paquetes.....	13
Figura 8 Pantalla: Inicio de sesión (dispositivo móvil y ordenador)	14
Figura 9 Entorno de pruebas abordado	15
Figura 10 Respuesta de una prueba en Postman.....	16
Figura 11 Revisión sistemática de los distintos surveys	19
Figura 12 Gráfica que jerarquiza el número de veces que aparecen las categorías en los artículos analizados.....	19
Figura 13 Gráfica ilustrativa que muestra las aplicaciones más relevantes en función de su categoría	20
Figura 14 Estructura del demostrador	21
Figura 15 Pantalla: herramienta de Big Data específica	22
Figura 16 Cronograma del proyecto	29
Figura 17 Desglose de tareas.....	29
Figura 18 Diagrama ilustrativo de la metodología de desarrollo abordada	30
Figura 19 Diagrama Kanban	31
Figura 20 Referencias analizadas	32
Figura 21 Gráfica ilustrativa que muestra las aplicaciones más relevantes en función de su categoría	32
Figura 22 Combinación de tecnologías utilizadas.....	38
Figura 23 Diagrama de módulos: Cliente	45
Figura 24 Diagrama de módulos: Servidor	46
Figura 25 Diagrama de despliegue.....	48
Figura 26 Diagrama Entidad-Relación.....	52
Figura 27 Fragmento de código que permite la sincronización entre el motor y la BD	53
Figura 28 Muestra las aplicaciones existentes en ElasticSearch	53
Figura 29 Muestra las aplicaciones existentes en la Base De Datos	53
Figura 30 Muestra el generador de consultas proporcionado por Dejavu	54
Figura 31 Muestra los resultados devueltos por ElasticSearch.....	54
Figura 32 Fragmento de código que protege las rutas de accesos sin sesión activa	55

Figura 33 Fragmento de código que verifica la existencia de una sesión activa	55
Figura 34 Respuesta del servidor para un acceso de sesión de un administrador.....	55
Figura 35 Fragmento de código que habilita la invocación de una funcionalidad sólo al administrador	56
Figura 36 Fragmento de código que habilita la invocación de una funcionalidad a todos los usuarios	56
Figura 37 Fragmento de código que recopila los datos para el informe.....	57
Figura 38 Fragmento de código que convierte los datos en formato CSV	57
Figura 39 Fragmento de código que permite la descarga de los informes	57
Figura 40 Fragmento de código que configura las opciones de las gráficas	58
Figura 41 Diagrama de secuencia: Registro.....	59
Figura 42 Diagrama de secuencia: Inicio de Sesión.	59
Figura 43 Diagrama de secuencia: Perfil de Usuario.....	60
Figura 44 Diagrama de secuencia: Actualización de Credenciales.....	60
Figura 45 Diagrama de secuencia: Valorar una aplicación.	61
Figura 46 Diagrama de secuencia: Comentar una aplicación.	61
Figura 47 Diagrama de secuencia: Crear Categoría.....	62
Figura 48 Diagrama de secuencia: Modificar Categoría.	62
Figura 49 Diagrama de secuencia: Borrar Categoría.....	63
Figura 50 Diagrama de secuencia: Crear Aplicación.	63
Figura 51 Diagrama de secuencia: Modificar Aplicación.....	64
Figura 52 Diagrama de secuencia: Borrar Aplicación.	64
Figura 53 Diagrama de secuencia: Borrar Cuenta.....	65
Figura 54 Diagrama de secuencia: Obtener Categorías	65
Figura 55 Diagrama de secuencia utilizando ElasticSearch: Obtener Aplicaciones de una categoría.	66
Figura 56 Diagrama de secuencia utilizando el Servidor: Obtener Aplicaciones de una categoría.	66
Figura 57 Diagrama de secuencia: Obtener todas las aplicaciones.....	67
Figura 58 Diagrama de secuencia: Descargar Informes.	67
Figura 59 Mapa de navegación para dispositivos móviles o similares	68
Figura 60 Mapa de navegación para ordenadores: Usuario no registrado.....	69
Figura 61 Mapa de navegación para ordenadores: Usuarios registrado.....	69
Figura 62 Mapa de navegación para ordenadores: Administrador.....	70
Figura 63 Pantalla: Home.....	71
Figura 64 Pantalla: Origen.....	71
Figura 65 Pantalla: Objetivos.....	72

Figura 66 Pantalla: Registro.....	72
Figura 67 Pantalla: Inicio de Sesión.....	73
Figura 68 Pantalla: Categorías existentes	73
Figura 69 Pantalla: Aplicaciones de una categoría	74
Figura 70 Pantalla: Herramienta concreta.	74
Figura 71 Pantalla: Estadísticas.	75
Figura 72 Pantalla: Perfil de Usuario.	75
Figura 73 Pantalla: Panel de Administrador.....	76
Figura 74 Pantalla: Creación/Activación encuestas	76
Figura 75 Pantalla: Creación de preguntas.....	77
Figura 76 Pantalla: Responder Encuesta	77
Figura 77 Pantalla: Encuesta Desactivada	78

1. Introducción

TECNARA es una entidad sin ánimo de lucro fundada por la Asociación de Empresarios de Informática de Aragón (AEIA) el 4 de febrero de 1988. Esta asociación representa, promociona y defiende los intereses de las empresas informáticas de Aragón. Por lo tanto, entre sus objetivos principales destaca ayudar en la transformación digital de la Comunidad de Aragón, promoviendo el desarrollo de nuevas competencias y tecnologías. Actualmente, uno de los aspectos tecnológicos más demandados en esta comunidad, está relacionado con lo que se denomina Big Data. Es en este contexto en el que surge el proyecto Formatea, el cual formará a 121 jóvenes aragoneses en el sector TIC, cuyos únicos requisitos será no poseer empleo actual y tener entre 16 y 30 años. A continuación, explicaremos el origen de este proyecto, además de los problemas que intenta solventar y los objetivos de este.

1.1. Aragón y el Big Data

Aragón posee una situación geográfica privilegiada, ya que ofrece una perfecta comunicación tanto con las principales ciudades españolas, como un rápido flujo de comunicación con el resto de Europa. Características claves, que tal y como muestra un estudio (*Aragón, 2019*) de la estructura económica de Aragón, contribuye favorablemente al sector dedicado a la industria.

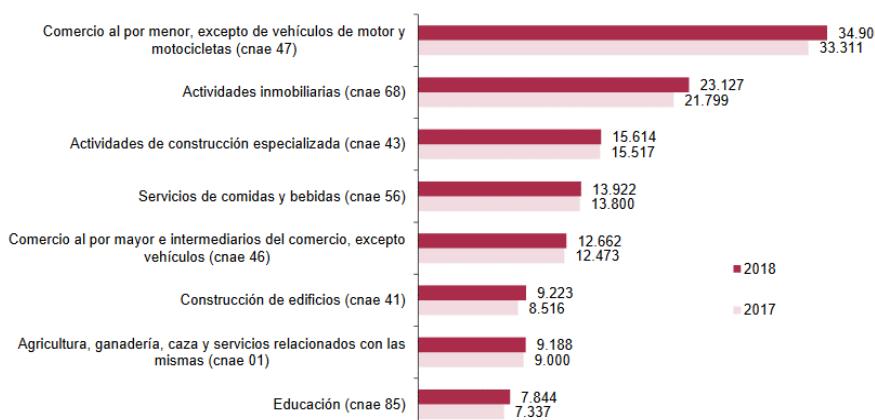


Figura 1 Ranking de actividades económicas de Aragón

Dentro del sector de la industria, destacan subsectores como automoción, logística o industria agroalimentaria, los cuales hacen un elevado uso de herramientas de Big Data, para estudiar procesos, interconectar vehículos o estudiar el comportamiento del cliente entre otros. Por esa razón, el uso de este tipo de herramientas en esta comunidad ha crecido exponencialmente, lo cual se puede ver ratificado en titulares como:

- PSA Firma Un Convenio Con El CAAR Para Impulsar La Innovación (*Llorente, 2019*)

- Aragón Aplicará Big Data e IA Para La Predicción y Control de Plagas En El Viñedo (*Serrano, 2019*)

Pero ¿sabemos qué significa el término Big Data para las empresas aragonesas? TECNARA sospecha que las empresas aragonesas tienen una interpretación parcial del concepto de Big Data, ya que estas lo asocian únicamente a volumen y, ciertamente, sí intenta resolver esa problemática del manejo de grandes cantidades de datos, pero abarca mucho más que eso. De hecho, Big Data se suele definir como “conjunto de técnicas que permiten analizar, procesar y gestionar conjuntos de datos extremadamente grandes que pueden ser analizados informáticamente para revelar patrones, tendencias y asociaciones” (*Diccionario de Español jurídico, 2020*). Además, el volumen no tiene definido un tamaño mínimo que divida, lo que es Big Data y lo que no. Según, IBM (*Ricardo Barranco Fragoso, 2012*), no existe una cantidad de datos específica, aunque afirma que usualmente se habla en términos de petabytes y exabytes de datos. Véanse ejemplos comprensibles para una empresa aragonesa:

- **Gigabyte**: equivale aproximadamente a 256 canciones. Debido a que el tamaño promedio de cada canción es igual a 4 MB.
- **Terabyte**: : cantidad equivalente a 4 portátiles de 256 GB, teniendo en cuenta que el S.O. ocupa parte de ese espacio.
- **Petabyte**: todas las fotos que posee Facebook equivalen a 1.5 PB (*Iregui, 2011*). Dicha cantidad no es tratada en empresas aragonesas.
- **Exabyte**: dicha unidad tampoco es gestionada en las empresas de Aragón, pero para hacernos una idea 15 EB es el número total de datos que tiene Google repartido entre todos sus servidores (*Santos, 2018*). Google, Amazon o Facebook son las únicas empresas que manejan tales cantidades de datos.

Entonces, ¿qué es Big Data para las empresas aragonesas si no trabajan en niveles de petabytes, ni exabytes?

Nº	Review	Title					
			VELOCITY	VOLUME	VALUE	VARIETY	VERACITY
1	The Scientific World Journal	Big Data: Survey, Technologies, Opportunities, and Challenges	X	X	X	X	
2	Journal of King Saud University	Big DATA technologies: A survey	X	X		X	
3	Journal of Big Data	Big Data analytics: a survey	X	X	X	X	X
4	Journal of Big Data	A survey on platforms for big data analytics				X	
5	IEEE COMMUNICATIONS	Deep Learning for IoT Big Data and Streaming Analytics: A Survey	X	X	X	X	X
6	Journal of Big Data	A survey of open source tools for machine learning with big data in the Hadoop ecosystem	X	X		X	
7	International Journal	A Survey on Big Data Analytics: Challenges, Open Research Issues and Tools	X	X	X	X	X
8	University of St Andrews	Undefined By Data: A Survey of Big Data Definitions	X	X		X	X
9	IEEE Internet Computing	From Databases to Big Data	X	X		X	
10	IEEE Internet Computing	Biology: The big challenges of big data	X	X		X	X
11	Semantic Scholar	Big Data: A Survey					
12	IEEE Internet Computing	A Survey of Clustering Algorithms for Big Data: Taxonomy and Empirical Analysis	X	X		X	
13	IEEE Internet Computing	In-Memory Big Data Management and Processing: A Survey	X		X		X

Figura 2 Análisis de las características que delimitan Big Data

Para responder esta pregunta utilizaremos la imagen anterior que muestra un análisis realizado en este TFG sobre distintos surveys¹, los cuales coinciden en que su comportamiento no solo está definido por **volumen**, sino que también está definido por **variedad, velocidad, valor, visibilidad y/o veracidad** (*Mohammadi et al., 2018*) como atributos determinantes que muestran si una empresa utiliza Big Data o no. Veamos un ejemplo de una empresa que conozco directamente, por mi actual trabajo en ella, el departamento de infraestructura de IT en Opel España, la cual está orientada al sector de la automoción. ¿Cuáles de las "V" del Big Data me he encontrado en ella?

- **Volumen:** se relaciona con el crecimiento de los datos y la tasa de desarrollos de las unidades de disco. Puedo confirmar que no hay volumen de datos suficiente.
- **Variedad:** viene detallado por el tipo de datos y su forma de agrupación o categorización para manejar diferentes tipos de datos (numéricos, categóricos y jerárquicos). En Opel, la variedad del dato es importante ya que maneja diferentes tipos de datos, entre los que destacan empleados, piezas, proveedores o el cliente al que va destinado el vehículo, entre muchos otros. En media cada vehículo tiene 60.000 componentes y la mayoría de estos tienen un seguimiento individualizado por vehículo fabricado (más de 600.000 vehículos/año)
- **Velocidad:** relaciona la velocidad de un algoritmo de agrupamiento en cuanto a la complejidad y el rendimiento de este, ya que una gran cantidad de datos son requeridos en tiempo real. En Opel, el proceso de montaje completo de cada vehículo es de aproximadamente 20 horas, pero en la línea de montaje se ensamblan una media de 57 vehículos por hora, es decir aproximadamente un coche cada minuto, por lo que la velocidad es una característica clave. En resumen, esto indica que miles de sensores han capturado información del proceso de producción de los 60.000 componentes de cada vehículo por minuto y se han almacenado en un "data center"².
- **Veracidad:** propiedad que recoge la calidad, consistencia y confiabilidad del conjunto de datos estudiados. En Opel, a lo largo del ciclo de construcción del coche le acompaña un "data center", el cual comunica al robot las piezas, colores o características que debe incorporar a la carrocería, por lo que esa información debe ser fiel a la elección establecida por el cliente.
- **Variabilidad:** característica dependiente a la tasa de frecuencia de datos. Existen aplicaciones en las que la frecuencia de recogida de datos varía lo suficiente para realizar picos cargantes. Este tema se puede asociar con la salida de los coches Bitono en la planta de Figueruelas, los cuales requieren realizarse en dos fases de pintado con un

¹ Añadido al diccionario de términos

² Añadido al diccionario de términos

descanso de 24 horas entre ambas fases. A mayor demanda de este tipo de vehículos, mayor congestión se produce en los almacenes, por lo que los sistemas deben adaptarse a la demanda realizada, ya que no habrá dos "data centers" iguales en los vehículos producidos.

- **Valor:** se define como la transformación de datos en información útil, capaz de producir una ventaja organizativa. Estos se obtienen de procesos o servicios subyacentes. El valor de los datos es clave en este sector, un ejemplo de ello es la obtención de los niveles de carbono emitidos por los vehículos, pudiendo tener consecuencias legales negativas en caso de no obtener los valores válidos. Al igual que los componentes, estos niveles serán guardados en el "data center" que acompaña al vehículo durante todo su ciclo de producción.

Es decir, en mi experiencia personal me he encontrado con todas las "V" menos con la "V" que habitualmente asociamos con Big Data. Esta impresión también la comparte TECNARA, ya que piensa que el sector tecnológico aragonés no aprovecha bien el Big Data, debido a las "falsas creencias" de utilizar este tipo de herramientas solo con grandes cantidades de datos.

1.2. Problemas abordados

El principal problema de TECNARA es el desconocimiento del entorno local de Big Data. Debido a que existe un gran abanico de herramientas existentes, es necesario acotarlo para orientar cursos como Formatea a las herramientas que más se adecúan a las necesidades de las empresas. A su vez aprovecharán esta oportunidad para realizar una doble maniobra, ya que además de obtener un portal que permita descartar las herramientas menos relevantes de Big Data, obtendrán un portal genérico que permita a las empresas compartir su opinión sobre el uso y/o valoración sobre cualquier aspecto a estudiar. Además, debido a las dificultades provocadas por la pandemia del COVID-19, al sistema se le añadió la funcionalidad de realizar encuestas con el fin de recopilar información sobre las formaciones telemáticas, entre otras cuestiones.

1.3. Objetivos del proyecto

El principal objetivo, consiste en la construcción de una plataforma denominada "TECNARA Tools", la cual permite a las empresas compartir información sobre el objeto a estudiar y, así, generar una serie de informes agregados que faciliten la toma de decisiones de diferentes estudios. Posteriormente, adecuar dicho sistema a un escenario concreto, es decir, el Big Data, de manera que permita a TECNARA conocer las herramientas de manejo de datos más demandadas y las dificultades que afronta cada compañía. Consecuentemente, el clúster podrá ofertar una formación óptima, adecuando dichas formaciones a las necesidades de

cada empresa, ya que se obtendrá una clara visualización del perfil laboral, que necesitan/necesitarán en un futuro próximo el sector tecnológico de Aragón.

Como objetivos personales en relación con el proyecto, me gustaría perfeccionar el manejo de algunas de las herramientas utilizadas para la construcción de esta aplicación y adquirir nuevos conocimientos sobre las herramientas ofertadas en estos cursos, con el fin de poderlos aplicar en mi actual ámbito laboral, al mismo tiempo que aprovecho la oportunidad para conocer y relacionarme con distintos profesionales del sector estudiado.

1.4. Organización y gestión del proyecto

Para la elaboración de dicho proyecto, se ha creado un plan de gestión, acorde a sus dos temáticas principales. El primero de sus bloques se centra en un plan, que garantice la construcción en tiempo y modo del portal. Este bloque abarca las siguientes fases descritas a continuación:

- **Análisis inicial y captura de requisitos:** analiza el entorno y recopila los requerimientos que debe soportar el sistema para dar solución al problema.
- **Diseño e implementación de la plataforma web:** modela y crea el sistema.

Por el contrario, el segundo bloque, se basa en un plan que garantice la creación de una instancia del sistema construido anteriormente. Esta instancia está enfocada al Big Data y será un demostrador de lo que se puede hacer con “Tecnara Tools”, consiguiendo una serie de informes que faciliten la toma de decisiones referentes al proyecto Formatea. Entre sus fases destacan las siguientes:

- **Revisión sistemática de la literatura:** fase de síntesis, que delimita los conceptos de Big Data y obtiene un conjunto de herramientas de código abierto en lo referente al manejo de datos, para la posterior carga del demostrador.
- **Creación del demostrador (plataforma + contenidos) y carga de datos inicial:** fase establecida para crear la instancia que permite equipar de datos óptimos el sistema.
- **Evaluación del demostrador:** fase de evaluación del demostrador elaborado.

Ambos planes de gestión poseen un proceso de evolución basado en tareas, por lo que la elaboración del proyecto se realizará secuencialmente por prioridad de requisitos y no se avanzará al siguiente sin la consecución del anterior. Finalmente, se realizó también tanto una planificación temporal del proyecto como la especificación del flujo de comunicación que se iba a llevar a cabo. Estos detalles, se pueden observar en el apartado “*Anexo B Gestión del proyecto*”.

1.5. Estructura de la memoria

La memoria del proyecto está organizada en cuatro partes, además de secciones complementarias, como bibliografía y anexos. El primer bloque o introducción, detalla el enfoque general de la investigación, ubicando la problemática, definiendo los objetivos a alcanzar y especificando la gestión establecida para afrontar el proyecto. El segundo bloque se centra en el proceso llevado a cabo para la creación y elaboración de un portal genérico, el cual permite a las empresas aragonesas compartir información sobre el uso y la utilidad de un objeto a estudiar. El tercer bloque define el proceso que ha permitido la obtención de un conjunto de herramientas de código abierto de Big Data, consiguiendo la construcción de un demostrador óptimo que adapta el sistema al dominio estudiado. Finalmente, el último bloque está destinado a obtener una serie de conclusiones y analizar el grado de consecución del proyecto.

2. TECNARA Tools

Este bloque define el proceso para la creación del sistema “TECNARA Tools”. Un portal que permite compartir información y valorar tanto el uso como la utilidad que aporta el objeto estudiado a cada empresa de manera concreta. Además, este mismo portal permitirá la descarga de una serie de informes de las contribuciones que han realizado las distintas empresas. Finalmente, debido a las dificultades provocadas por la pandemia del COVID-19 y la necesidad de conocer el pensamiento de las distintas empresas sobre las formaciones telemáticas, entre otras cuestiones, se añadió la funcionalidad de creación de encuestas.

2.1. Requisitos iniciales

“TECNARA Tools” debe aportar una visión genérica de las herramientas de libre pago más utilizadas, gracias a la colaboración de las distintas instituciones asociadas. Para llegar a ese cometido, tanto “TECNARA Tools”, como el demostrador de “TECNARA Tools” deben cubrir los requerimientos iniciales propuestos por TECNARA. Entre los requisitos iniciales del sistema destacan:

Código	Descripción
RS1	La plataforma debe permitir solo el acceso a cuentas que representen empresas o instituciones
RS2	La plataforma debe permitir a las empresas compartir información
RS3	La plataforma debe permitir valorar el objeto cargado
RS4	La plataforma debe mostrar informes basados en la información aportada por las distintas empresas
RS5	La plataforma debe permitir la realización de encuestas
RS6	La plataforma debe poseer un diseño web apto para todo tipo de dispositivos.

Por otro lado, entre los requerimientos iniciales del demostrador, destacan los siguientes:

Código	Descripción
RD1	El demostrador debe recoger las distintas categorías de herramientas de Big Data
RD2	El demostrador debe contener el conjunto de aplicaciones clasificadas en categorías, según su funcionalidad
RD3	El demostrador debe trabajar con las principales herramientas de Big Data utilizadas en la actualidad

De ese modo, el demostrador final permitirá a las empresas valorar y contribuir añadiendo o restando valor a las herramientas que utilizan y el motivo por el que hacen uso de estas, independientemente de los dispositivos que dispongan. A continuación, otras compañías podrán comentar estas opiniones. Para dar validez a la información aportada, las empresas deberán registrarse para asegurarnos que son usuarios relacionados con cuentas institucionales, es decir, empresas y organizaciones afiliadas a TECNARA. Este sistema, además, habilitará a TECNARA la recogida de una serie de resultados, los cuales facilitarán la toma de decisiones y adicionalmente, se permitirá la realización de encuestas para corroborar estos resultados.

Con la consecución de un sistema que realice los requisitos mostrados anteriormente, TECNARA podrá favorecer positivamente al sector tecnológico de Aragón, mediante la realización de cursos, como por ejemplo los focalizados en las herramientas de Big Data punteras.

2.2. Análisis del sistema

Este proyecto debe obtener una serie de informes lo suficientemente robustos, como para soportar la toma de decisiones de uno de los proyectos de mayor prioridad de TECNARA, ya que han recibido una elevada cuantía por ello. Por lo tanto, el portal tiene que facilitar la utilización de este, siendo confortable su navegación, ya que el número de las intervenciones por parte de las empresas es clave en este proyecto. Además, debe cumplimentar los potenciales casos de uso, los cuales permitirán reducir el gran abanico de cursos que ofrecer. Estos son:

- **Crear cuenta:** caso de uso que comienza cuando un usuario desea registrarse.
- **Buscar aplicación:** caso de uso por el cual un usuario puede encontrar el elemento concreto sobre el cual desea realizar una acción.
- **Comentar aplicación:** caso de uso que permite añadir/modificar o borrar una vivencia o experiencia sobre un determinado elemento.
- **Valorar aplicación:** caso de uso que permite añadir o quitar valor sobre un determinado objeto.
- **Responder encuestas:** caso de uso que permite responder una serie de encuestas, con el fin de afianzar los resultados obtenidos.

Tras examinar los principales casos de uso que se debían implementar, se propusieron dos alternativas en lo referente a la inserción de aplicaciones. La primera de ellas planteaba, que estas aplicaciones fueran añadidas por las distintas empresas. En contraposición, surgía la idea de que estas fueran insertadas por TECNARA. Finalmente, se optó por esta segunda

opción, debido a la gran cantidad de aplicaciones, las cuales no son utilizadas por el sector tecnológico de Aragón, por lo que el resultado no sería un entorno representativo.

Respecto a las tecnologías que darían forma al portal, se evaluaron dos posibles opciones en la parte del cliente, Angular y Vue.js. Ambas facilitan desarrollar aplicaciones a gran escala y poseen documentación con gran cantidad de ejemplos. Finalmente, Angular fue la elegida por haber sido utilizada previamente. En la capa de lógica no hubo duda, siempre se quiso utilizar Node.js debido a la comodidad del lenguaje y la experiencia que se tenía previamente sobre él, lo cual disminuiría notablemente el tiempo del proyecto y, por tanto, se conseguiría la consecución de la entrega del proyecto en tiempo.

2.3. Usuarios del sistema

Previo a describir los requisitos finales, se identificaron y describieron en colaboración con TECNARA los diferentes usuarios que iban interaccionar con el sistema. Por lo tanto, se procede a definir a los diferentes usuarios implicados:

- **Clientes:** Usuarios en representación de empresas, que hacen uso de dicha web para verificar las herramientas, que más se adaptan a su necesidad.
- **Colaboradores:** Usuarios en representación de empresas, que aportan la experiencia previa obtenida con dichas tecnologías y contribuyen de manera indirecta, para fomentar a nuevos trabajadores.
- **Administradores del sistema:** Usuarios que actúan con un rol de gestor, sobre las cuentas de usuario, creación de aplicaciones, encuestas, descargas de informes, etc.

Respecto a los usuarios actuando como clientes/colaboradores, se podrían definir cómo el mismo perfil de usuario, pero se ha decidido realizarlo de esta manera por realizar una distinción entre las funcionalidades ofertadas.

2.4. Casos de uso

A continuación, mostramos los diferentes diagramas de casos de uso, en los cuales se sustenta el portal implementado.

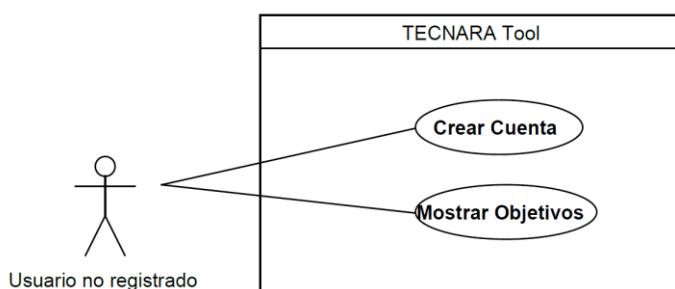


Figura 3 Casos de uso: Usuario no registrado

Usuario no registrado:

- **Crear Cuenta:** comienza cuando un usuario desea registrarse. Este posee dos flujos alternativos. El primero, si el sistema reconoce el dominio del email como una institución asociada, lo que quiere decir que está incluido en la entidad “dominios” de la Base de Datos. En este caso, solo tendrá que activar el registro desde su correo. El segundo, en caso de no ser reconocido, será el administrador el que decida si habilita su acceso.
- **Mostrar Objetivos:** presenta ante el cliente una lista, la cual muestra los principales objetivos del sistema.

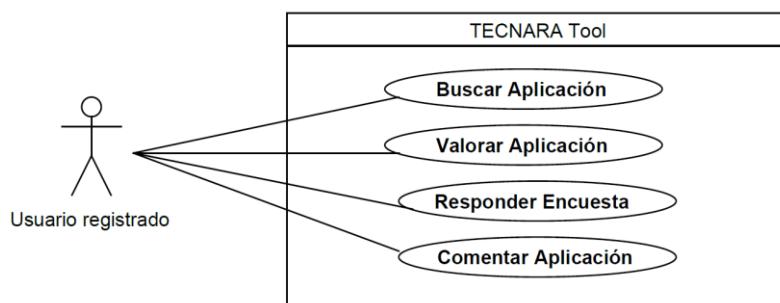


Figura 4 Casos de uso: Usuario registrado

Usuario registrado:

- **Buscar Aplicación:** permite encontrar una aplicación concreta, mediante distintos criterios de búsqueda.
- **Comentar Aplicación:** permite añadir/modificar o borrar una vivencia o experiencia sobre un determinado elemento. Se desea que estos comentarios solo puedan ser borrados o modificados por el propio usuario durante el primer día posterior a su inserción, para no ver influenciados sus pensamientos.
- **Valorar Aplicación:** permite añadir o quitar valor sobre una determinada herramienta. Un usuario solo podrá votar una vez sobre una determinada herramienta.
- **Responder Encuestas:** permite responder una serie de encuestas con el fin de afianzar los resultados obtenidos. Estas encuestas pueden ser desactivadas para modificar el tiempo disponible de respuesta.

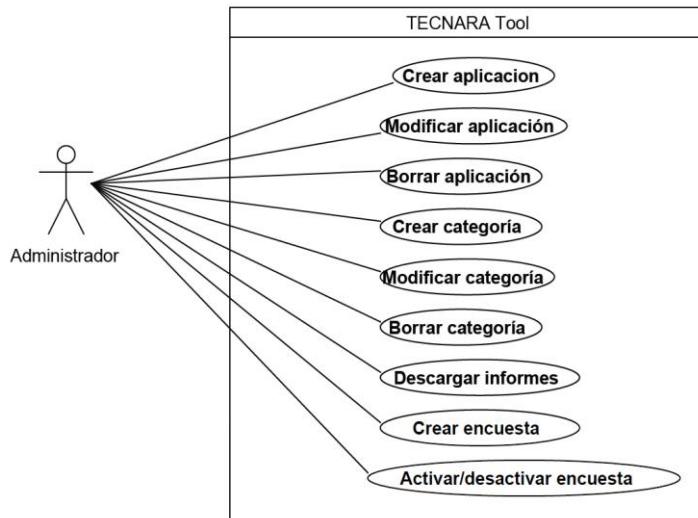


Figura 5 Casos de uso: Administrador

Administrador:

- **Crear/modificar/borrar aplicaciones:** permite crear, modificar o borrar una nueva aplicación al sistema creado.
- **Crear/modificar/borrar categorías:** permite crear, modificar o borrar una categoría que recoja una serie de aplicaciones.
- **Descargar informes:** permite descargar los informes obtenidos de las diferentes contribuciones de las empresas.
- **Crear encuestas:** crear una serie de encuestas, con el fin de estudiar un problema concreto. Estas encuestas podrán dedicarse a grupos concretos, los cuales estarán formados por usuarios registrados. Además, se podrá limitar el tiempo de respuesta de estos.
- **Activar/desactivar encuestas:** caso de uso que permite limitar el tiempo de respuesta de estas y orientar la encuesta al conjunto de usuarios deseado.

Para verificar los detalles concretos sobre los casos de uso al completo, consultar el apartado “Anexo G Diagramas de Caso de Uso”.

2.5. Requisitos identificados

Los requisitos funcionales del sistema “TECNARA Tools”, vienen relacionados por los casos de uso que debe sostener el sistema y los cuales han sido explicados anteriormente. Además, como el portal debe ofertar distintas funcionalidades, según el rol del usuario, se ha distinguido la especificación de los requisitos, en función del actor que la puede llevar a cabo (U: usuario, A: administrador). En lo que respecta a la lista de requisitos al completo, esta puede ser consultada en la sección “E Requisitos de la aplicación”.

Entre los requisitos funcionales más importantes, recalcaría la gestión del acceso al portal, los diferentes métodos de búsqueda de aplicaciones, las propiedades características de una contribución sobre una determinada herramienta o la gestión de encuestas.

En lo referente al acceso al portal, solo se debe permitir a usuarios que actúen en representación de una empresa o institución. Seguidamente, la búsqueda de herramientas concretas se podrá realizar en función de distintos criterios, cómo categoría, nombre o valoración entre otros. Posteriormente a ser encontrada y tras realizar una contribución en una determinada herramienta, esta aportación sólo podrá ser modificada o borrada en un corto periodo de tiempo, para evitar opiniones influenciadas. Finalmente, sobre la gestión de encuestas, estas podrán ser creadas solo por el administrador y podrán ser limitadas a responder en el tiempo que este considere oportuno. Además, las podrá orientar a grupos específicos, los cuales poseen un conjunto de usuarios editable.

Por el contrario, entre sus requisitos no funcionales yo remarcaría, que la interfaz del sistema debe ser capaz de adaptarse a todos los dispositivos independientemente del tamaño de estos, es decir, debe poseer un diseño *responsive*. Finalmente, se cree conveniente la descarga de informes en formato CSV, los cuales ilustran las contribuciones realizadas por las empresas en torno al objeto estudiado, y así, aprovechar su formato columnar y la versatilidad que posee este formato con herramientas de análisis como Power BI³.

2.6. Diseño del sistema

Posteriormente a la identificación de los requisitos que TECNARA deseaba abordar, se diseñó un diagrama arquitectural lógico que mostraba el comportamiento del sistema a desarrollar.

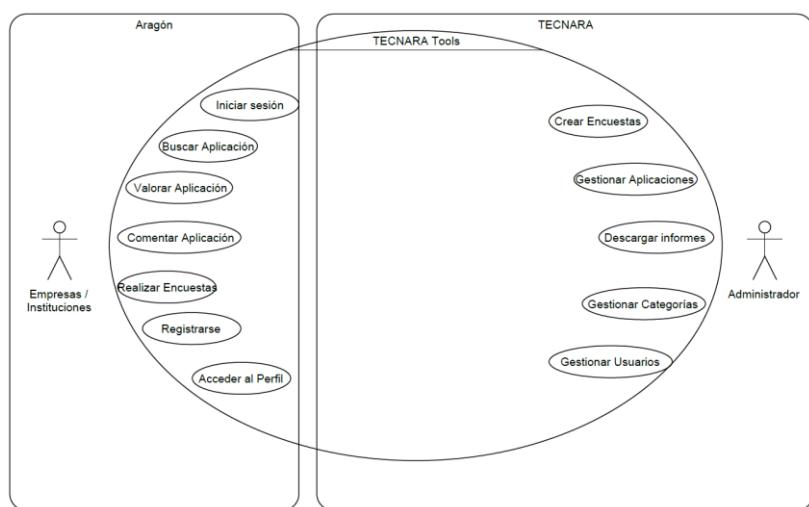


Figura 6 Diagrama arquitectural lógico

³ Añadido al diccionario de términos

Este esquema realizaba una primera división en función de la procedencia del usuario. En primer lugar, Aragón, componente que agrupa al conjunto de empresas, los cuales podrán realizar los casos de uso ilustrados en la parte izquierda del diagrama, tales como: registrarse, iniciar sesión, modificar el perfil creado, buscar un objeto específico, compartir información, establecer una valoración o responder una serie de encuestas. Por el contrario, Tecnara, agrupa a un usuario especial, el administrador. Este podrá realizar además de los casos de uso comentados anteriormente, los ilustrados en la parte derecha del diagrama, tales como: gestión de usuarios, creación de encuestas, gestión de tanto aplicaciones, como categorías, verificación de estadísticas o descargar informes representativos.

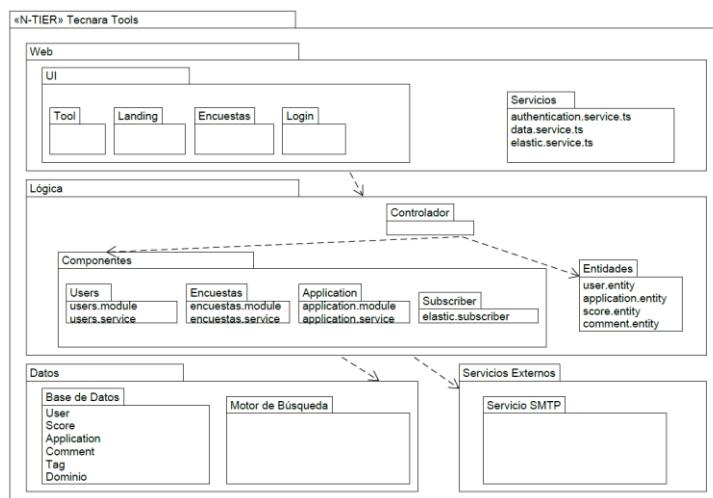


Figura 7 Vista de paquetes

La arquitectura mostrada anteriormente y en la que se asienta el sistema, es una arquitectura basada en capas. Estas capas son: web, lógica y datos. La web, es la capa apreciable por el usuario y la cual retransmite la información a la siguiente capa. Para ello, utiliza: UI⁴, capa compuesta por componentes formados por una parte visual que será implementada en HTML y CSS y una parte funcional implementada en TypeScript. Además de una capa que recoge los servicios, los cuales invocan tanto a las funciones ofertadas por la parte lógica, como las ofertadas por un motor de búsqueda. El siguiente nivel, la capa lógica, posee tres componentes: el controlador: encargado de transmitir las peticiones del cliente con el componente adecuado, las entidades: representan a los distintos registros de la base de datos y los componentes: importan las librerías a utilizar e implementan la lógica de la aplicación. Finalmente, destaca tanto una capa externa, constituida por un servidor SMTP para el envío de emails, como una capa de datos que recoge tanto a la Base de Datos, con las tablas que esta posee, como el motor de búsqueda, el cual posee índices asociados a las categorías del

⁴ Añadido en acrónimos

objeto cargado (Ej: Framework, Query...) y documentos asociados al propio elemento cargado (Ej: Hadoop, Spark...).

2.7. Implementación

La implementación del proyecto está asentada por la arquitectura del sistema mostrada anteriormente y las secciones descritas en “Anexo H Vista de módulos”, y “Anexo J Modelado de datos”. En cuanto a la metodología de implementación de las distintas instancias con el fin de reducir el tiempo de despliegue, se ha realizado con Docker⁵, utilizando las siguientes imágenes: Portainer (gestión de contenedores), Dejavu (interfaz gráfica para el motor de búsqueda), Postgres (base de datos) y ElasticSearch (motor de búsqueda). Cabe destacar, que el proceso se realizaba en función de la prioridad de los requisitos que se deseaban completar, por lo que la estructura de estas instancias fue el primer paso a seguir. Debido a su prioridad, se implementaron primero los requisitos que permitían el acceso al portal.

El acceso al portal es un escenario que debe ser capaz de realizar un filtrado, permitiendo el acceso únicamente a cuentas institucionales. Debido a esto, no todos los usuarios que deseen registrarse van a ser admitidos. Esta admisión se va a tratar de dos formas complementarias basadas en un algoritmo de registro sujeto a confirmación. El funcionamiento de ambas funcionalidades se apoya en un servidor SMTP, el cual mediante una expresión regular obtiene los dominios de registro del correo proporcionado y realiza una comparación con los dominios conocidos por TECNARA (@unizar.es, @opel-vauxhal.es, @everis.es...). En caso de no coincidir, el usuario estaría pendiente a que el administrador acepte/descarte dicha petición. Por el contrario, si coincide, el usuario estaría únicamente pendiente de la activación de su cuenta mediante la confirmación de un email, el cual le redireccionaría a la pestaña de inicio de sesión adaptada al tamaño del dispositivo que el usuario esté utilizando.

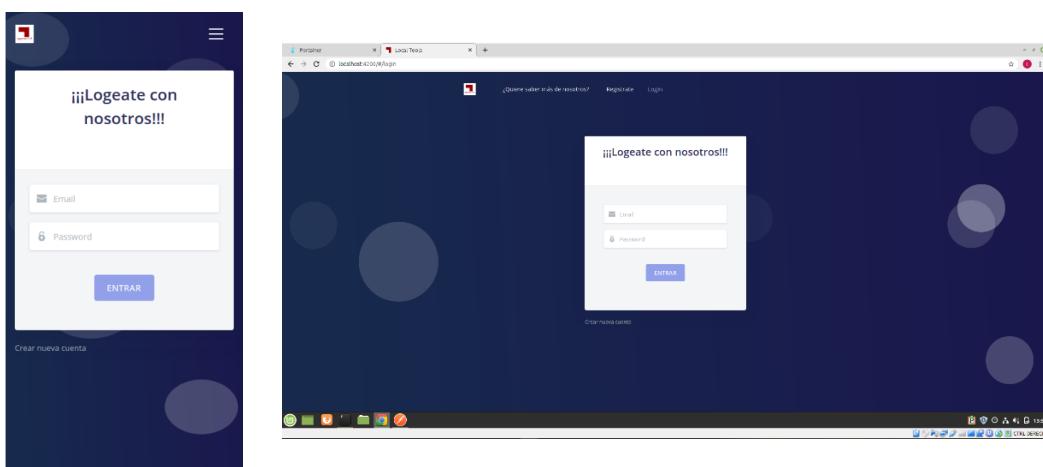


Figura 8 Pantalla: Inicio de sesión (dispositivo móvil y ordenador)

⁵ Añadido al diccionario de términos

Seguidamente, se implementaron las funcionalidades que afectaban directa/indirectamente a las aplicaciones, tales como, la visualización de un catálogo estructurado, el proceso de valoración, la gestión de comentarios o la creación de encuestas.

La visualización del catálogo es un punto relevante, ya que el conjunto de aplicaciones supera el medio centenar de herramientas. La solución que se propuso fue agruparlas en categorías que delimitaran su marco de extensión. Todo ello, sustentado por el motor de búsqueda ElasticSearch, detallado en la sección “*Anexo J.1 Sincronización entre Elasticsearch & BBDD*”. De esta manera, el motor permite ofrecer a los usuarios distintos criterios de búsqueda, como búsquedas por valoración o por nombre, entre otras. El proceso de valoración solo debe permitir la inserción de valor una única vez en cada aplicación. Dicha restricción, estaba contemplada en el diseño, ya que la entidad puntuación tenía relación con usuario y aplicación, lo que permitía realizar restricciones. En lo que concierne al borrado o modificación de comentarios, fueron delimitados por un periodo temporal fijo, para evitar conductas influenciadas. Esto se consiguió mediante la comparación de la fecha de inserción y la fecha en la que se desea actualizar. Una funcionalidad extra que aportaba solidez a los informes generados se basaba en las encuestas. Estas encuestas debían poder ser dirigidas a subconjuntos específicos de usuarios, por ello se utilizó un servidor SMTP, el cual redistribuía la encuesta al grupo seleccionado gracias a la entidad asociación. Además, el tiempo de respuesta debía poder ser limitado, por lo que el administrador podía desactivarla, y no permitir la inserción de nuevas respuestas.

Finalmente, se habilitó al administrador una pestaña para verificar los resultados obtenidos. Además, estos informes también pueden ser descargados, como indica la sección “*Anexo J.5 Descargar informes*”, lo cual facilita el análisis y la versatilidad, ya que permite integrarlo con herramientas como Power BI⁶.

2.8. Pruebas

A este proyecto también se le ha otorgado de un período de tiempo dedicado exclusivamente a pruebas. Este tipo de pruebas, tienen como objetivo principal, la verificación del sistema elaborado, el cual debe asegurar la realización de todos sus requisitos funcionales de manera adecuada. Para ello se planificó un escenario que sustentara dichas acciones:

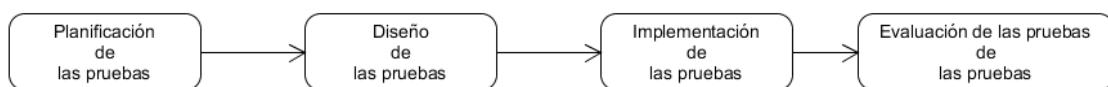


Figura 9 Entorno de pruebas abordado

⁶ Añadido al diccionario de términos

En rasgos generales, debido a que el desarrollo del sistema se basaba en la consecución de requisitos de manera secuencial, estos eran testeados de manera progresiva e independiente. Consecuentemente, tras finalizar el último requisito, se realizó un segundo testeo general, para complementar los problemas previos encontrados. Estas pruebas se pueden dividir en la capa web y capa lógica.

- **Capa web:** esta parte ha sido testeada por TECNARA, tanto la parte estética y visual, como la comprobación de todas las invocaciones proporcionadas. Para ello, clientes potenciales que utilizarán esta aplicación en un futuro, han validado desde una perspectiva objetiva, la satisfacción con el diseño elaborado.
- **Capa lógica:** a medida que se iban implementando nuevas funcionalidades, estas iban siendo probadas con Postman⁷. De esta manera, se confirmaba la obtención de los resultados y respuestas esperadas, otorgándole a estas respuestas códigos de error, los cuales facilitaran la captura de estos en la parte web. El conjunto de estas pruebas ha sido exportado como un proyecto, incluyendo cada una de las funcionalidades ofertadas. Prueba de ello es la imagen mostrada a continuación, la cual ilustra el comportamiento implementado por el portal para el acceso de usuarios.

Figura 10 Respuesta de una prueba en Postman

En resumen, el proceso de pruebas ha solventado una serie de problemas, la mayoría de estos relacionados con evitar acciones inesperadas por parte de los distintos usuarios, tales como textos vacíos o la obtención de información de distinta manera a la esperada. Además, se incrementó la realimentación mostrada al usuario, para facilitar la usabilidad del sistema.

⁷ Añadido al diccionario de términos

3. TECNARA Tools aplicado a un caso: Big Data

3.1. Caso de aplicación

El sistema, como ya se comentó, se creó de manera genérica para que este pudiera ser utilizado en otros estudios, independientemente del dominio estudiado. Centrándonos en el caso presente, la aplicación requiere ser cargada en torno a un dominio específico, el Big Data. Es por ello por lo que este sistema necesita una carga propicia, la cual dote de sentido el software creado. Con el fin de obtener ese demostrador representativo de lo que se puede hacer con “TECNARA Tools”, se ha definido una estrategia, la cual permite la creación de una instancia compatible con el sistema.

3.2. Estrategia

La estrategia llevada a cabo, para la obtención de una carga correcta capaz de ilustrar el conjunto de herramientas de Big Data de código abierto disponibles, es un proceso que abarca distintas fases diferenciadas. La primera de ellas consiste en una etapa de creación de contenidos, la cual genera una materia prima apta, gracias a la revisión sistemática de distintas fuentes. Posteriormente, “TECNARA Tools” se personaliza y se le añaden herramientas específicas, que TECNARA desea incluir en el portal creado. Finalmente, se necesitan dos fases para completar el proceso, la creación de una instancia con los datos obtenidos y la fase que realiza la carga de la instancia creada.

3.3. Creación de contenidos

3.3.1 Definición de la pregunta de investigación y sus objetivos

El marco general de la investigación se enfoca hacia el conocimiento del área de estudio en el que está sumergido el proyecto, obteniendo una imagen ilustrativa de las herramientas de Big Data que existen en la actualidad y pueden ser útiles para las empresas aragonesas. Proporcionando de esta manera, una carga de datos coherente y óptima al portal anteriormente desarrollado. Además, coexiste con el objetivo descrito previamente otras metas cómo enriquecer personalmente el lenguaje tecnológico

3.3.2 Metodología

Para poblar nuestro portal de herramientas de Big Data, correctamente clasificadas y poder obtener una información fiable a través de la revisión de los informes del portal, se propuso la realización de una revisión sistemática basada en una búsqueda SMS⁸ o Systematic Mapping Study⁹. Este tipo de revisión sistemática ha sido elegida debido a que interesa la cantidad de evidencias estudiadas, el tópico a tratar es amplio y está basado en un proceso que permite

⁸ Añadido en acrónimos

⁹ Añadido en el diccionario de términos

la realización de búsqueda en anchura, permitiendo un análisis superficial de las distintas fuentes de información. Este estudio viene motivado por el auge de descubrir qué herramientas son frecuentemente más utilizadas en los proyectos de Big Data según los distintos *surveys*¹⁰.

El proceso de trabajo se basa en el análisis de distintas fuentes de información, obtenidas mediante la utilización del repositorio Google Scholar u otras herramientas que proporciona la Universidad de Zaragoza. Respecto al patrón de búsqueda escrito en los distintos repositorios web, siempre ha sido “*Big Data Survey*”. Cada fuente de información obtenida era evaluada, es decir incluida o excluida. Estas causas de inclusión (I) o exclusión(E), de cada resultado obtenido se evaluaba considerando su título, abstract, palabras clave, extensión o el tipo de acceso. Viéndose delimitado por los siguientes criterios:

- **I1:** el resultado es una revisión o *survey* de trabajos anteriores sobre Big Data.
- **E1:** no está escrito en inglés.
- **E2:** su extensión es inferior a cinco páginas.
- **E3:** el acceso exige pago.
- **E4:** es un resultado duplicado o redundante.
- **E5:** el trabajo describe en su mayoría aspectos conceptuales, sin concretar tecnologías.
- **E6:** el trabajo es un whitepaper comercial.

Respecto la metodología de extracción de datos, se aprovecha constantemente la experiencia previa de búsquedas piloto, ya que tanto las fuentes de información analizadas como la tabla proporcionada por FirstMark (Turck, 2018) están basados en estudios previos, los cuales trabajan con un conjunto de herramientas de Big Data de código libre. El proceso de extracción se ha complementado con la creación de una lista de acrónimos y una descripción de las herramientas que se han adjuntado a esta memoria. Finalmente, se verifica en los distintos *surveys* obtenidos la aparición y el contexto en el que aparecen las distintas tecnologías comentadas.

Debido a la pretensión que es un tema relevante y existen gran cantidad de *surveys* publicados, el criterio de parada establecido inicialmente fueron 25 artículos de inclusión. A continuación, se muestra un ejemplo ilustrativo del procedimiento realizado.

¹⁰ Añadido en el diccionario de términos

Nº	Revista	Título	Framework												Query / dataflow							
			Hadoop (HDFS)	Hadoop MapReduce)	Apache Kylin	Slider	Yarn	Flink	Spark	Mesos	CDAP	GridGain	Apache Hama	Avro	Spart SQL	Presto	Sqoop	Pig	Hive	Apache Drill	Google Cloud Dataflow	Flink
1	The Scientific World Journal	Big Data: Survey, Technologies, Opportunities, and Challenges	X	X										X				X	X			
2	Journal of King Saud University	Big DATA technologies: A survey	X	X		X	X	X						X				X	X	X		
3	Journal of Big Data	Big Data analytics: a survey	X	X										X					X	X		
4	Journal of Big Data	A survey on platforms for big data analytics	X	X		X	X												X	X		
5	IEEE COMMUNICATIONS	Deep Learning for IoT Big Data and Streaming Analytics: A Survey	X	X				X	X													
6	Journal of Big Data	A survey of open source tools for machine learning with big data	X	X		X	X	X						X				X	X	X	X	X
7	International Journal of Big Data Engineering and Management	A Survey on Big Data Analytics: Challenges, Open Research Problems and Future Trends	X	X															X	X		
9	IEEE Internet Computing	From Databases to Big Data	X	X																		
10	IEEE Internet Computing	Biology: The big challenges of big data		X																		
11	Health AFFAIRS	Big Data And New Knowledge In Medicine: The Thinking, The Tools, And The Questions		X																		
13	IEEE Internet Computing	A Survey of Clustering Algorithms for Big Data: Taxonomy and Empirical Analysis																				
14	IEEE Internet Computing	In-Memory Big Data Management and Processing: A Survey	X	X																		
19	Harvar University	What Big Data Reveal About Survey Misreporting and the Real Electorate																				
20	Journal on Advances in Big Data	A survey of machine learning for big data processing	X	X														X	X			
29	International Journal of Big Data	(A Survey On Big Data Analytics In Health Care																				
30	International Institute for Business Administration	Big Data in Big Companies		X	X														X	X		
31	Frontiers of Computer Science	Big data challenge: a data management perspective																				
34	Journal of Engineering Research	Different Techniques to Transfer Big data: a Survey																				
35	International Journal of Big Data	A review paper on big data and Hadoop	X	X										X						X		
37	Journal of Computer Science	Survey of Large-Scale Data Management Systems for Big Data	X	X										X						X		
41	International Journal of Big Data	Recent advances delivered by Mobile Cloud Computing and Internet of Things for Big Data applications: a survey																				
44	International Journal of Big Data	Solutions for RDBMS Problems -ASurvey	X	X																		
45	PLOS Computational Biology	Unmet needs for analyzing biological big data: A survey of 704 NSF principal investigators																				
51	Indian Journal of Science and Technology	A Survey on Clustering Techniques for Big Data Mining	X																X	X		
54	Imperial College of Engineering	Survey Paper On Big Data	X	X				X	X									X	X			

Figura 11 Revisión sistemática de los distintos surveys

3.3.3 Resultados

Entre las conclusiones obtenidas procedentes de la revisión sistemática, destaca el alargamiento de los ciclos de vida del dato (recolección, filtrado, análisis, almacenamiento, publicación, recuperación y finalmente pérdida de este), lo cual genera dificultad en el manejo de estos. Es por esto por lo que, entre las categorías existentes, existe una que aborda mejor esta problemática comentada, la categoría de Framework¹¹. Categoría que se reafirma ya que es la categoría que recoge mayor número de apariciones en los surveys analizados, seguidas por Query, Data Access o Streaming.

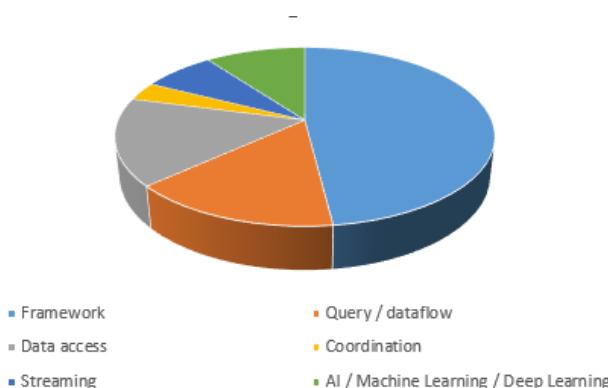


Figura 12 Gráfica que jerarquiza el número de veces que aparecen las categorías en los artículos analizados

¹¹ Añadido en el diccionario de términos

En lo que concierne a las aplicaciones, se puede reflejar la rápida evolución del entorno empresarial ya que la mayoría de los surveys analizados tienen un período de vida relativamente corto, un claro indicador de la gran aclimatación que está teniendo el sector tecnológico con estas herramientas. En el interior de Framework, categoría que más ha destacado, resalta la aplicación de Hadoop (HDFS y MapReduce)¹², ecosistema que fue creado para utilizar únicamente datos de internet y que finalmente evolucionó para analizar y procesar grandes cantidades de datos, tanto estructurados como no estructurados.

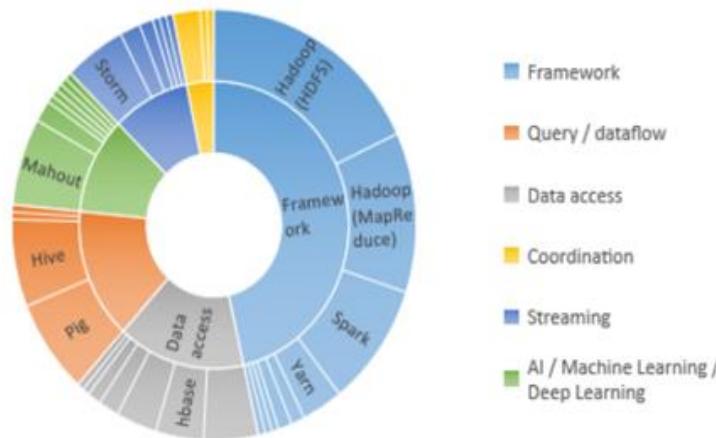


Figura 13 Gráfica ilustrativa que muestra las aplicaciones más relevantes en función de su categoría

El fabricante que cuenta con mayor un número de herramientas de las analizadas es Apache. Entre sus aplicaciones destacan: Accumulo, Ambari, Avro, Beam, Drill, Flink, Flume, Yarn, Giraph, Oozie, Spark, Storm, Sqoop o Hadoop entre muchas otras¹³.

Tras finalizar el proceso de revisión sistemática, se obtiene una primera carga con el contenido de las aplicaciones y sus respectivas categorías, que era el objetivo primordial. Además, se tiene una primera idea de las aplicaciones que pueden ser las más valoradas por el sector tecnológico y por tanto obtener una mayor aceptación de los cursos en la comunidad de Aragón. Previamente a construir la instancia, que permitiría llevar a cabo la carga se realizó una reunión con todo el equipo para debatir si era una carga óptima o no, e incluir los deseos de aplicaciones concretas que el Clúster deseaba añadir. La valoración fue positiva y se adentró en la siguiente fase, fase centrada en la elaboración de los diferentes ficheros JSON que realizarían ese cometido.

¹² Definiciones añadidas en diccionario de términos

¹³ Herramientas añadidas en diccionario de términos

3.4. Diseño del demostrador

El demostrador debe ser capaz de cubrir los requerimientos mostrados con el prefijo “RD” en la sección “2.1 Requisitos iniciales”. Además, hay que tener en cuenta que el conjunto de datos obtenido tras realizar la revisión sistemática era diverso y de gran tamaño. Es por ello por lo que se acordaron con TECNARA los detalles que debía poseer tanto cada herramienta de Big Data como la categoría que las recogía, con el fin de poder obtener un punto de vista adecuado por parte de las distintas empresas.

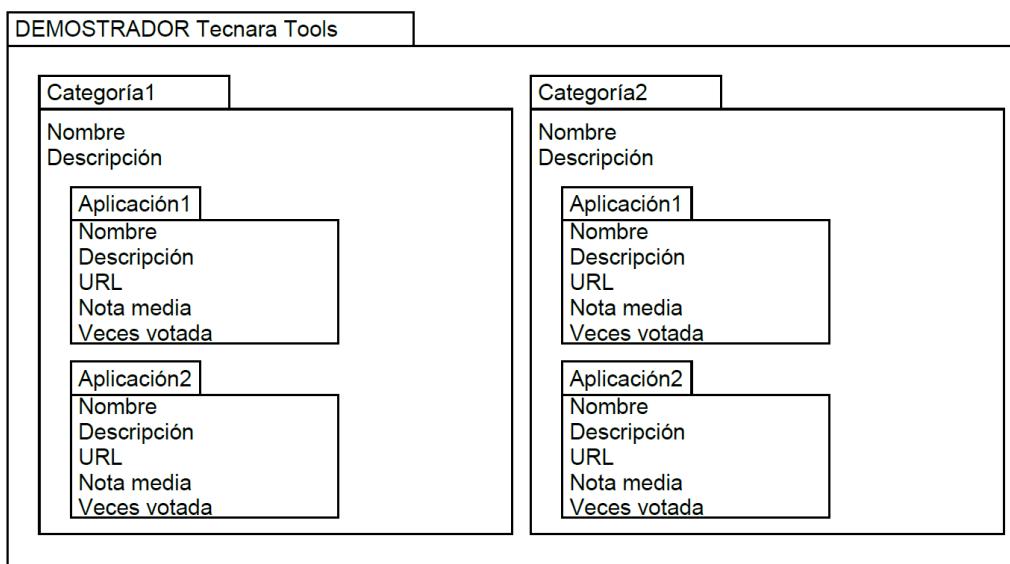


Figura 14 Estructura del demostrador

Como muestra el esquema anterior, el demostrador está sustentado gracias a categorías, los cuales recogen tanto los documentos relacionados entre sí, como una pequeña descripción de sí mismas. Por otra parte, los documentos estarán basados en las herramientas de Big Data, por ello contendrán el nombre, la descripción y una imagen representativa. Además, también contendrán la nota media de las valoraciones obtenidas y el número de votaciones recogidas.

3.5. Despliegue y carga de datos

El proceso de carga se realizó mediante las funciones proporcionadas por la API, con el fin de mantener en armonía los dos componentes de la capa inferior. Esta carga debe ser realizada de manera secuencial debido a que el motor de búsqueda lo requiere. Se necesita realizar la carga de los índices, previo a introducir los documentos que se indexarán. En este caso las claves se asemejan a nuestras categorías y los documentos a nuestras aplicaciones. Es decir, la primera carga de datos se realiza mediante el fichero “indices.json” y la segunda con el fichero “aplicaciones.json”. Estas cargas han sido realizadas de manera progresiva debido a requerimientos internos tales como aspectos de visualización.

Finalmente, el demostrador ha sido desplegado por la compañía APSER¹⁴, y está siendo usado por TECNARA, para decidir el conjunto de cursos de Big Data que ofertarán.

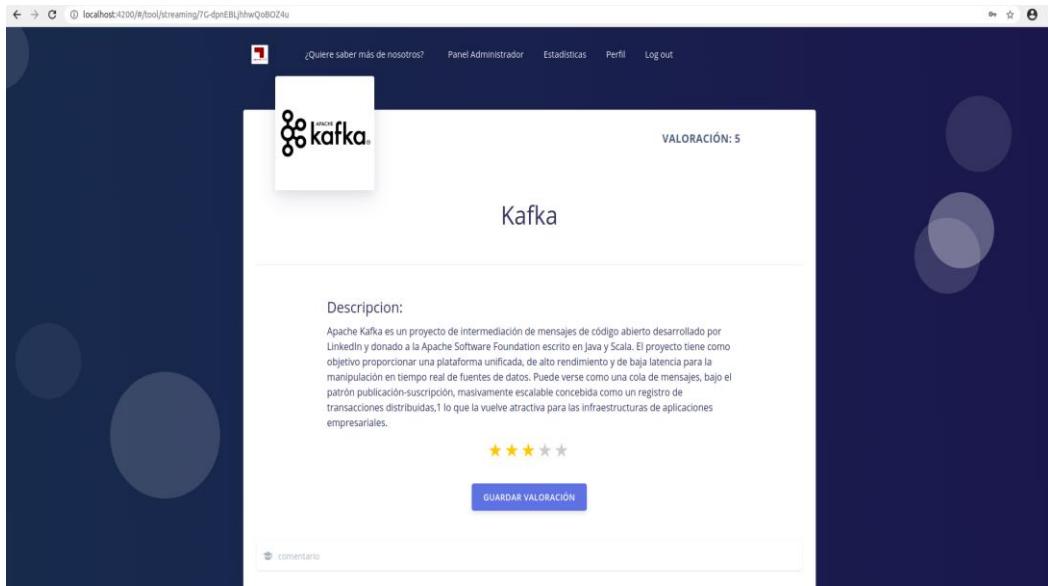


Figura 15 Pantalla: herramienta de Big Data específica

¹⁴ Empresa catalana que presta servicios TIC de alta disponibilidad a pequeñas empresas en materia de integración de datos o alojamiento de aplicaciones de alta disponibilidad.

4. Conclusiones

En este apartado, se recogen las distintas conclusiones obtenidas, tras la consecución del proyecto. El proyecto presente, además de la elaboración de un portal genérico denominado “TECNARA Tools” que permitiera compartir información entre las distintas instituciones, sugería adaptar el sistema a una temática concreta construyendo un demostrador específico de lo que se puede hacer con “TECNARA Tools”. Debido a estas razones, las conclusiones han sido divididas en: sistema, demostrador, aprendizaje y líneas futuras. Bloques que muestran las deducciones obtenidas al elaborar el sistema, el demostrador, lo aprendido a lo largo del proyecto y el futuro del sistema construido, respectivamente.

4.1. Sobre TECNARA Tools

A lo largo de la construcción del sistema, he afianzado mis conocimientos sobre las tecnologías utilizadas para la implementación del portal, ya que tanto Angular como Node.js, habían sido empleadas en asignaturas cursadas previamente. Adicionalmente, he utilizado librerías que nunca había empleado, tales como, la librería ElasticSearch, la cual facilita el flujo de comunicación con el motor de búsqueda, la librería Echart, la cual facilita la presentación de distintos gráficos, o Nodemailer, el cual permite el envío de emails a distintos usuarios. Al realizar el sistema, también he aprendido a utilizar una herramienta de Big Data, la cual desconocía por completo. Esta herramienta, denominada ElasticSearch, destaca por realizar búsquedas rápidas en grandes volúmenes de datos y permitir desacoplar las operaciones CRUD del servidor, además, este nuevo conocimiento adquirido lo he podido integrar en un proyecto de mi actual puesto laboral. Finalmente, aunque ya había utilizado diferentes contenedores de Docker, también he afrontado el riesgo de la utilización de contenedores, los cuales nunca había usado. Entre ellos destacan, Dejavu¹⁵ que proporciona una interfaz cómoda para la utilización de Elastic, que a la vez es otro contenedor utilizado y Portainer¹⁶, contenedor usado para la gestión de los distintos contenedores.

4.2. Sobre la construcción del demostrador de Big Data con TECNARA tools

La elaboración del demostrador ha incrementado enormemente mis conocimientos, debido a que el proyecto requería también saber ilustrar el entorno de tecnologías que iban a dotar de sentido el portal. Además, era novato en la realización de revisiones sistemáticas y estudios del arte, pero estos eran necesarios para obtener un conjunto de herramientas actual, variado

¹⁵ Añadido al diccionario de términos

¹⁶ Añadido al diccionario de términos

y completo, con el cual las empresas pudieran valorar o compartir información de interés de estas.

En conclusión, el demostrador incluye un conjunto de herramientas que representan una imagen fiel a las herramientas utilizadas por las empresas de Aragón en torno al Big Data.

4.3. Sobre mi aprendizaje

A lo largo del proyecto ha quedado claro, que existe una gran variedad de herramientas software orientadas a Big Data, las cuales facilitan al manejo y análisis de datos, durante todo su ciclo de vida. Debido a que el análisis de datos puede proporcionar ventajas organizativas, me parece importante la formación sobre este tipo de herramientas, para su mejor explotación.

Respecto al trabajo realizado y su adecuación a los resultados creo que ha sido casi completo. El proyecto ha sido desarrollado eficazmente, en sus dos partes diferenciadas, tales como el sistema íntegro en sí, cumplimentando todos los requisitos solicitados por TECNARA y la adecuación del sistema en un demostrador específico. En la actualidad se encuentra en una fase de obtención de resultados, de manera que posteriormente el clúster los analizará y decidirá la serie de cursos que desea impartir. Además, se ha dejado totalmente documentada la estructura y funciones de cara a un posible incremento de funcionalidades orientada a futuros marcos de trabajo.

Personalmente, la experiencia de abordar este trabajo en un ámbito laboral como TECNARA me ha permitido obtener una experiencia personal y profesional muy satisfactoria, además de haber podido establecer relación con profesionales de distintas empresas, lo cual puede favorecerme profesionalmente en un futuro. Finalmente, haber podido utilizar una de las herramientas estudiadas (ElasticSearch) en mi actual puesto laboral (Opel España, Grupo PSA) ha sido totalmente satisfactorio.

4.4. Sobre el futuro de TECNARA Tools

El grado de versatilidad de este sistema es amplio, ya que este puede ser cargado y explotado en diferentes estudios, es decir, si TECNARA desea realizar futuras investigaciones enfocadas a otros ámbitos, puede reutilizar este portal para ello. En adición, debido al comportamiento empresarial de TECNARA, siendo una empresa que afronta nuevos proyectos en relativamente cortos períodos de tiempo, sé que, si necesitan ayuda a la toma de decisiones en algún nuevo proyecto, no dudarán en hacer uso del sistema.

Bibliografía

- Appbaseio. (2019). *appbaseio/dejavu - Docker Hub*. Docker Hub.
<https://hub.docker.com/r/appbaseio/dejavu/>
- Aragón, G. de. (2019). Actividades económicas en el territorio. *Instituto Aragonés de Estadística, cnae 49*.
https://www.aragon.es/documents/20127/1909615/COMUNICADO_actividades+economicas18.pdf/2c3c1699-3ef9-bb46-13be-21ccb172d60a?t=1570086407615
- Diccionario de Español jurídico. (2020). *Definición de big data - Diccionario del español jurídico - RAE*. <https://dej.rae.es/lema/big-data>
- Docker Official Images. (2019a). *ElasticSearch*. Docker Hub.
https://hub.docker.com/_/elasticsearch
- Docker Official Images. (2019b). *postgres* - Docker Hub. Docker Hub.
https://hub.docker.com/_/postgres
- Iregui, L. (2011). ¿Cuánto es un petabyte? Enter Co. <https://www.enter.co/otros/cuanto-es-un-petabyte/>
- Llorente, M. (2019). *PSA firma un convenio con el CAAR para impulsar la innovación*. El Heraldo. <https://www.heraldo.es/noticias/economia/2019/07/23/psa-firma-un-convenio-con-el-caar-para-impulsar-la-innovacion-1326744.html>
- Mohammadi, M., Al-Fuqaha, A., Sorour, S., & Guizani, M. (2018). Deep learning for IoT big data and streaming analytics: A survey. *IEEE Communications Surveys and Tutorials*, 20(4), 2923–2960. <https://doi.org/10.1109/COMST.2018.2844341>
- Portainer. (2018). *portainer/portainer* - Docker Hub. Docker Hub.
<https://hub.docker.com/r/portainer/portainer/>
- Portal CTA. (2020a). *BIG DATA ARCHITECTURE WORKSHOP | INAEM*. INAEM. <https://inaem.aragon.es/190999108-big-data-architecture-workshop>
- Portal CTA. (2020b). *Buscador de Cursos CTA | INAEM*. INAEM. <https://inaem.aragon.es/buscador-de-cursos-cta>
- Portal CTA. (2020c). *CIENTÍFICO DE DATOS BIG DATA CLOUDERA | INAEM*. INAEM. <https://inaem.aragon.es/190999110-cientifico-de-datos-big-data-cloudera>
- Ricardo Barranco Fragoso. (2012). ¿Qué es Big Data? IBM DeveloperWorks.
<https://www.ibm.com/developerworks/ssa/local/im/que-es-big-data/index.html>
- Santos, M. (2018). ¿Qué podemos guardar en un Petabyte o Exabyte? HardZone.
<https://hardzone.es/2018/05/25/cuantos-archivos-caben-petabyte-exabyte/>
- Serrano, E. (2019). Aragón aplicará Big Data e IA para la predicción y control de plagas en el viñedo. El Economista.
<https://www.eleconomista.es/aragon/noticias/10188778/11/19/Aragon-aplicara-Big-Data-e-IA-para-la-prediccion-y-control-de-plagas-en-el-vinedo.html>
- Turck, M. (2018). *BIG DATA & AI LANDSCAPE*. FIRSTMARK. http://mattturck.com/wp-content/uploads/2018/07/Matt_Turck_FirstMark_Big_Data_Landscape_2018_Final.png

A Diccionario de términos

A.1 Acrónimos

AEIA: Asociación de Empresarios de Informática de Aragón.

API: Application Programming Interface / Interfaz de programación de aplicaciones.

CTA: Centro de Tecnologías Avanzadas.

INAEM: Instituto Nacional Aragonés de Empleo.

KISS: Keep It Simple and Stupid / Mantener Simple y Estúpido.

SMS: Systematic Mapping Study / Estudio Sistemático de Mapeo.

UI: User Interface / Interfaz de usuario.

A.2 Diccionario

Tecnologías de Big Data:

Apache Accumulo: es una tienda de valor de clave distribuida, ordenada y altamente escalable basada en Bigtable de Google.

Apache Ambari: es una infraestructura abierta para suministro, gestión y supervisión de clústeres de Apache Hadoop. Entre sus características principales destaca que ofrece un panel de control para vigilancia de la salud y el estado del clúster.

Apache Hadoop: framework de código abierto que permite el almacenamiento distribuido y el procesamiento de grandes conjuntos de datos en base a un hardware comercial. En otras palabras, Hadoop hace posible a las organizaciones obtener conocimiento rápidamente a partir de cantidades masivas de datos.

Avro: se encuentra dentro del procedimiento Hadoop Apache y es un marco de serialización de datos y llamadas de procedimiento remoto orientado a filas que utiliza JSON para definir tipos de datos y protocolos. La versión inicial es de 2009.

Beam: herramienta que ayuda a las empresas a gestionar sus activos, su inventario, sus operaciones de mantenimientos entre otras.

Drill: framework libre que permite trabajar en análisis interactivos de grupos de datos a gran escala. Su objetivo principal se encuentra en encontrar una alta escalabilidad en servidores.

Flink: marco de procesamiento de flujo de código abierto de baja latencia el cual ha sido desarrollado por Apache Software Foundation

Flume: Apache Flume es un servicio distribuido, fiable, y altamente disponible para recopilar, agregar, y mover eficientemente grandes cantidades de datos.

Giraph: proyecto de apache que desarrolla grafos respecto a grandes cantidades de datos. Este utiliza internamente Hadoop MapReduce.

H2O: entorno de modelos de análisis de datos y machine learning.

Hbase: base de datos distribuida no relacional de código abierto modelada a partir de Google BigTable y escrita en Java.

HDFS. sistema de archivos distribuido, escalable y portátil escrito en Java para el framework Hadoop.

Hive: software que forma parte del ecosistema de Hadoop. Este es utilizado para gestionar enormes datasets almacenados bajo el HDFS de Hadoop y realizar consultas sobre estos mismos

Mesos: administrador de Clúster de código abierto escrito en C++. Este permite utilizar el clúster para correr los framework sobre los cuales se ejecuta la aplicación, distribuyendo la carga entre los distintos servidores y evitando así sobrecargas.

Oozie: planifica workflows de manejo de jobs de Hadoop. Utilizando la herramienta Oozie se puede definir tanto el comienzo como el final de estos workflows utilizando variables parametrizadas.

Spark: conjunto de librerías para el desarrollo de aplicaciones web en Java, el cual se basa en Hadoop Map Reduce y que, principalmente, permite dividir o paralelizar el trabajo, ya que normalmente se instala en un clúster de máquina.

Sqoop: herramienta utilizada para la transferencia de datos desde / al sistema de archivos distribuido Hadoop. Sqoop permite una fácil importación y exportación de datos de almacenes de datos estructurados como bases de datos relacionales, almacenes de datos empresariales, y sistemas NoSQL. Sqoop también podrá exportar a cualquier base de datos relacional.

Storm: librería de Python para la gestión de objetos relacionales entre una o más bases de datos SQL y objetos Python

YARN: siglas cuyo significado en inglés son “otro negociador de recursos”, es una tecnología de administración de clústeres, cuya función principal es dividir las funcionalidades de la gestión de recursos y la supervisión de trabajos.

Otros términos:

Data center: se denomina centro de procesamiento de datos al espacio donde se concentran los recursos necesarios para el procesamiento de la información de un objeto determinado.

Dejavu: interfaz de usuario web de ElasticSearch. Esta interfaz proporciona actualizaciones en tiempo real, vistas filtradas, entre otras funcionalidades.

Diccionario de Español jurídico: diccionario resultante entre la fusión de la RAE (Real Academia Española) y la CGPJ (Consejo General del Poder Judicial).

Disco duro de Google: dispositivo de almacenamiento de datos que utilizaba Google para guardar sus datos.

Docker: facilita la creación de contenedores de distintas tecnologías software para que estos puedan ser desplegados y utilizados, en cualquier máquina que tenga instalado docker.

Framework: entorno de trabajo que estandariza conceptos, prácticas y criterios para enfocar un tipo de problemática particular, en este caso el manejo de gran cantidad de datos.

Hook: término que se asemeja al término castellano de *ganch* y el cual permite tener la información síncrona en dos o más sistemas. De manera que, todo lo que publiquemos en un repositorio se actualice en el resto.

MapReduce: modelo de programación para dar soporte a la computación paralela.

Portainer: imagen de docker, la cual aporta una interfaz de usuario que facilita la gestión de los distintos contenedores proporcionados por Docker.

Postman: herramienta para probar API web, la cual simula el comportamiento de un cliente realizando peticiones http/https.

Power Bi: servicio de análisis empresarial de Microsoft, el cual intenta proporcionar visualizaciones interactivas y capacidades de inteligencia empresarial.

Survey: estudio estadístico que intenta abarcar una visión amplia en un dominio concreto.

Systematic Mapping Study: revisiones sistemáticas, que destacan por la utilización de un mayor número de criterios de inclusión y está destinado a mapear temas, concretamente tendencias de investigación.

B Gestión del proyecto

Gestión temporal

En cuanto a la planificación temporal del proyecto establecida puede ser comprobada en el siguiente Diagrama de Gantt. En este diagrama destaca el color gris, dedicado para eventos finalizados (a día de la línea temporal roja) y el color azul dedicado para los eventos activos.

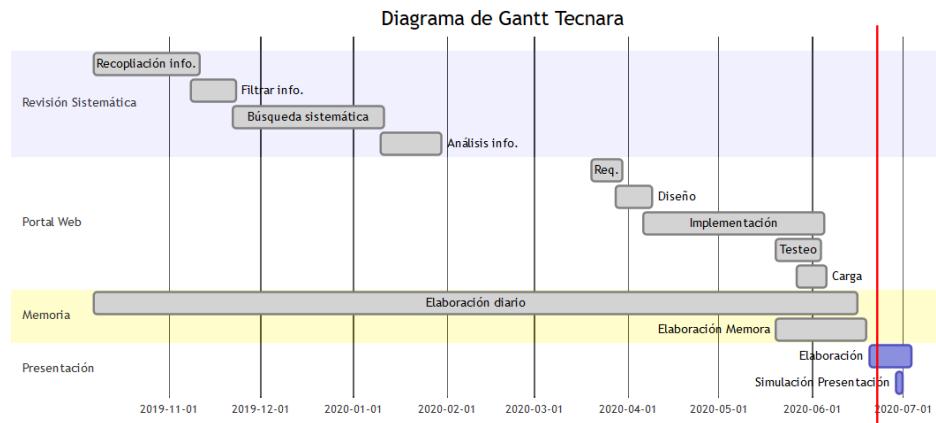


Figura 16 Cronograma del proyecto

Este diagrama de Gantt ilustra una primera división acorde a los puntos claves abordados en el proyecto, tales como: revisión sistemática, portal web, la memoria y finalmente la presentación asociada. Asimismo, con el fin de ilustrar más detalladamente la gestión de tiempo/trabajo se ha adjuntado el informe abordado durante toda la vida del proyecto.

Nombre	Fecha de inicio	Fecha de fin
Mantenimiento diagrama de Gantt	7/10/19	3/07/20
Revisión sistemática	7/10/19	29/01/20
Recopilación de información	7/10/19	12/11/19
Filtrado de información	8/11/19	22/11/19
Realización de búsqueda sistemática	22/11/19	15/01/20
Análisis de la información obtenida	10/01/20	29/01/20
Portal web	31/03/20	7/06/20
Recogida de requisitos	31/03/20	9/04/20
Análisis & Diseño	6/04/20	13/04/20
Implementación	13/04/20	7/06/20
Testeo	23/05/20	6/06/20
Carga del portal	27/05/20	31/05/20
Memoria	7/10/19	23/06/20
Elaboración de un diario con los procedimientos realizados	7/10/19	23/06/20
Elaboración de la memoria del proyecto	25/05/20	23/06/20
Presentación	20/06/20	3/07/20
Elaboración de la presentación	20/06/20	3/07/20
Simulación de presentación	29/06/20	30/06/20

Figura 17 Desglose de tareas

Metodología

El proyecto en su conjunto se adecúa a uno de los principios conceptuales de ingeniería del software, el principio KISS¹⁷ para facilitar la continuidad o modificación del sistema, en caso de que sea requerido de esta manera por parte de TECNARA. Además, se programaron reuniones intermedias destinadas a la verificación del proyecto, por parte de TECNARA. Dada la complejidad del producto y la necesidad de conseguir un demostrador óptimo en el menor tiempo posible, la metodología de desarrollo elegida ha sido el desarrollo en cascada secuencial con el fin de solventar los problemas en las primeras fases.

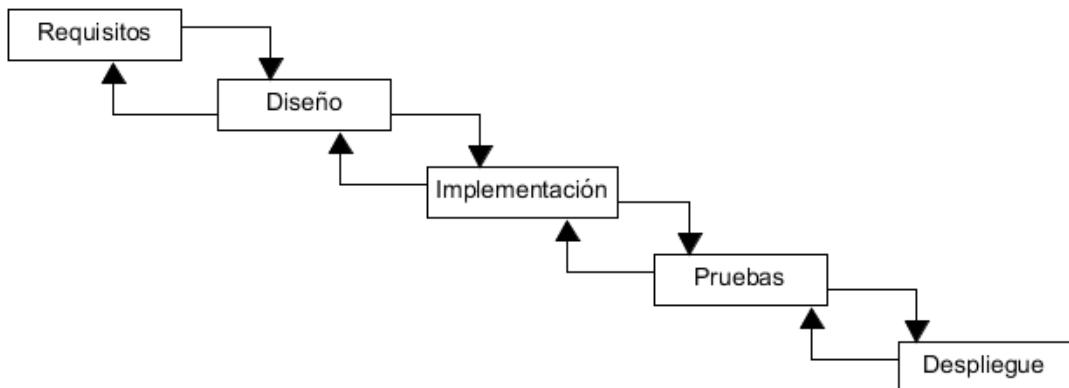


Figura 18 Diagrama ilustrativo de la metodología de desarrollo abordada

Comunicación

En lo que se refiere a la comunicación establecida, la primera parte del proyecto fue totalmente presencial, es decir se planificaron una serie de reuniones en las instalaciones de la universidad (EINA, Campus Río Ebro), con el fin de verificar el avance de la revisión sistemática de la literatura.

Seguidamente, tras la presentación con Jaime González Pina, director de este TFG y principal representante de TECNARA, se utilizaron las instalaciones del CEEI, para las reuniones semanales. Estas reuniones estuvieron dedicadas para recolección de requisitos, propuesta de diseño, verificar la implementación y realizar pruebas a distintos clientes potenciales. Adicionalmente, los avances técnicos han sido soportados por la herramienta de control de versiones gratuita de GitHub, en la cual se subían los distintos release y versiones de la aplicación y se comunicaban los nuevos requerimientos mediante una serie de templates adecuadas a nuestra necesidad:

¹⁷ Añadido al diccionario de términos

- **Issue report**: plantilla modificada con el fin de conseguir una serie de datos claves para focalizar y solventar el error.

- **Innovation request**: plantilla modificada para definir los nuevas funcionalidades que debían ser implementadas por el sistema.

Además, esta versión permite crear un repositorio privado accesible sólo a colaboradores del proyecto. En este caso, el acceso estuvo disponible durante todo el periodo de realización del proyecto para todos los integrantes del equipo. Es decir, tanto el director del proyecto, como el responsable de la parte técnica.

Durante los diferentes ciclos de vida del proyecto, a este se le ha acompañado con la utilización de la metodología Kanban. Metodología que ha sido dividida en dos proyectos totalmente diferenciados. El primero de ellos basado en el propio proyecto en sí (sistema y demostrador), gestionando las nuevas peticiones, errores reportados y las funcionalidades abordadas, las cuales estaban clasificadas en subtareas y añadidas a su fase correspondiente. El segundo y último, proyecto cuyo contenido residía en la elaboración de la memoria y su añadida presentación, en el cual cada uno de los apartados de esta se gestionaban como tareas independientes pasando por el estado “in process” cuando se estaba trabajando en ella y “done” cuando esta se había finalizado.

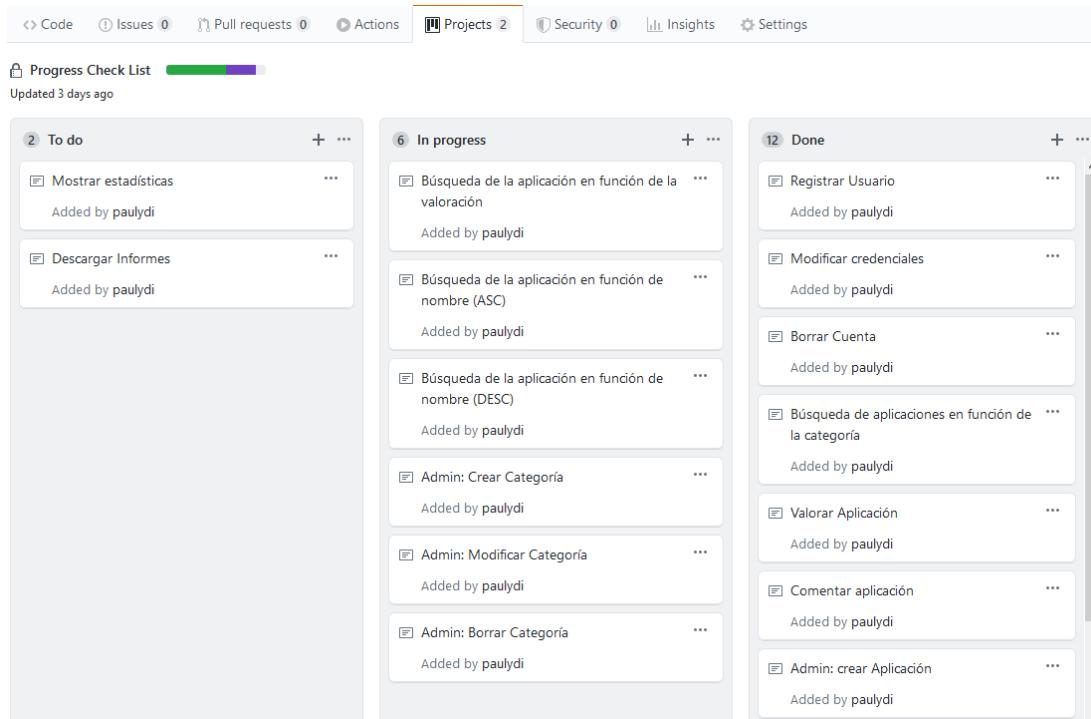


Figura 19 Diagrama Kanban

Finalmente, debido al tema de la pandemia de Covid-19, las reuniones eran telemáticas mediante Zoom con la misma periodicidad comentada anteriormente.

C Análisis sistemático

Este anexo pretende complementar los resultados obtenidos en la sección “3.3.3 Resultados”, sobre el análisis de los diferentes surveys¹⁸. En dicha sección, se mostraron gráficos con los resultados de las distintas categorías y aplicaciones. Por lo tanto, se procede a adjuntar una tabla de las referencias que han sido analizadas.

Nº	Revista	Título	Autores	Año	Inclusión	Enlace
1	The Scientific World Journal	Big Data: Survey, Technologies,	Nawsher Khan; Ibrar	2014	I1	ok https://www.hindawi.com/journals/tswj/2014/712826/
2	Journal of King Saud University - Com	Big DATA technologies: A survey	Ahmed Oussous, Fatima-Za	2018	I1	ok https://www.sciencedirect.com/science/article/pii/S13191
3	Journal of Big Data	Big Data analytics: a survey	Chun-Wei sai, Chin-Feng La	2015	I1	ok https://journalofbigdata.springeropen.com/articles/10.118
4	Journal of Big Data	A survey on platforms for big data analytics	Dilpreet Singh and Chandan	2014	I1	ok https://journalofbigdata.springeropen.com/articles/10.118
5	IEEE COMMUNICATIONS SURVEY	Deep Learning for IoT Big Data and Streaming	Mehdi Mohammadi, Ala Al	2018	I1	ok https://arxiv.org/pdf/1712.04301.pdf
6	Journal of Big Data	A survey of open source tools for machine learn	Sara Landset, Taghi M.Khos	2015	I1	ok https://journalofbigdata.springeropen.com/articles/10.118
7	International Journal of Advanced Com	A Survey on Big Data Analytics: Challenges, [Kauser Ahmed P., D. P. Ach	2016	I1	ok https://www.researchgate.net/profile/Debi_Acharya/public	
8	University of St Andrews	Undefined By Data: A Survey of Big Data Defi	Jonathan Stuart Ward and A	2013	E2	ok https://arxiv.org/pdf/1309.5821.pdf
9	IEEE Internet Computing	From Databases to Big Data	Sam Madden	2012	I1	ok https://ieeexplore.ieee.org/abstract/document/6188576
10	IEEE Internet Computing	Biology: The big challenges of big data	Vivien Marx	2013	I1	ok https://ieeexplore.ieee.org/abstract/document/7548112
12	Mobile Networks and Applications	Big Data: A Survey	Min Chen, Shiwen Mao, Yunf	2014	E3 - I1	Acceso desde UZ https://link.springer.com/article/10.1007/s11036-013-048
13	IEEE Internet Computing	A Survey of Clustering Algorithms for Big Data	Shiwen Mao	2014	I1	ok https://ieeexplore.ieee.org/abstract/document/6832486
14	IEEE Internet Computing	In-Memory Big Data Management and Proces	Yunhuo Liu	2015	I1	ok https://ieeexplore.ieee.org/abstract/document/7097722
15	Elsevier	Data-intensive applications, challenges, techn	Philip Chen	2013	E3 - I1	Acceso desde UZ https://www.sciencedirect.com/cuarczo/unizar.es.9443/sc
16	IEEE Internet Computing	Survey on NoSQL database	Jing Han, Haihong E., Guan	2011	E3 - E5	Enlace equivocado https://ieeexplore.ieee.org/cuarczo/unizar.es.9443/abstract
17	IEEE Internet Computing	Learning to Hash for Indexing Big Data—A Sur	Jun Wang, Wei Liu, Sanjiv K	2016	E3 - E5	Acceso desde UZ https://ieeexplore.ieee.org/cuarczo/unizar.es.9443/abstract
18	Journal of Business Research	Assessing business value of Big Data Analyti	Nadine Corte-Real, Tiago Oliveira	2017	E3 - E5	Acceso desde UZ https://www.sciencedirect.com/cuarczo/unizar.es.9443/sc
19	Political Analysis	What Big Data Reveal About Survey Misreport	Stephen Ansolabehere, Eliza	2012	I1	ok https://www.cambridge.org/core/services/app/cambridge-core
20	Journal on Advances in Signal Proces	A survey of machine learning for big data proc	Junfei Qiu, Qihua Wu, Guoru	2016	I1	ok https://asp-eurasipjournals.springeropen.com/articles/10
21	IEEE Internet Computing	Big data: A review	Seref Sagiroglu, Duygu Sina	2013	E3 - I1	Acceso desde UZ https://ieeexplore.ieee.org/abstract/document/6567202
22	Journal of Network and Computer App	A survey of machine learning for big data proc	Aisha Siddiqi, Ibrahim Abak	2016	E3 - i1	Acceso desde UZ https://www.sciencedirect.com/cuarczo/unizar.es.9443/sc
23	TDWI Research	Big Data Analytics	Philip Russom	2011	E6	Es un White paper https://womentre.com/wp-content/uploads/2016/04/big-da
24	Inder Science	Privacy models for big data: a survey	Nancy Victor, Daphne Lopez	2014	E3	ok https://www.inderscienceonline.com/doi/pdf/10.1504/IJBD
25	University Shenyang	Survey on NoSQL for Management of Big Dat	HEN De-Rong, YU Ge, WANG	2013	E3	ok http://en.cnki.com.cn/Article_en/CJFDTotal-RXB201308
26	Public Opinion Quarterly	Big Data in Survey Research: AAPOR Task Force	Marcus Berg, Paul Biemer	2015	E3 - I1	Acceso desde UZ https://academic.oup.com/cuarczo/unizar.es.9443/poq/ant
27	Public Opinion Quarterly	Feature selection methods for big data bioinfor	Lipo Wang, Yaoli Wang, Qin	2015	E3 - E5	Acceso desde UZ https://www.sciencedirect.com/cuarczo/unizar.es.9443/sc
28	ACM Digital Library	Bio data and cloud computing: current state	Dasarikanth Acaraju, Sudipto	2011	E3 - E2	Acceso desde UZ http://dl.acm.org/10.1145/1960000/1961432/oE30.w

Figura 20 Referencias analizadas

Las referencias obtenidas y analizadas ascienden a un número total de sesenta y dos. Este número de análisis realizados, superan el plan de calidad creado inicialmente donde se esperaban un número total de veinticinco referencias de inclusión estudiadas. Finalmente, con esta misma revisión se obtuvieron las aplicaciones más destacables para la población óptima del demostrador, en función de su categoría. A continuación, se muestra una gráficas con las aplicaciones mayormente destacadas en la categoría de Framework:

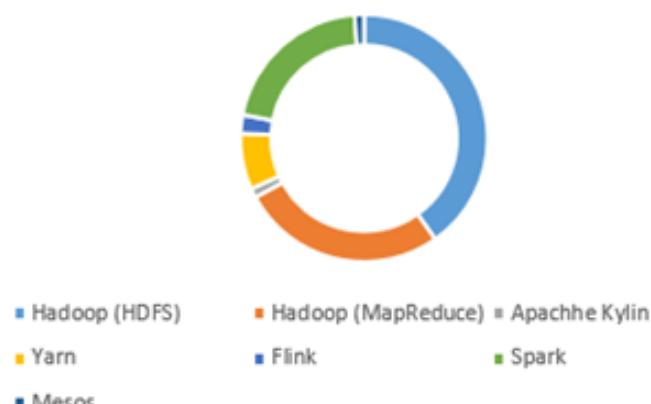


Figura 21 Gráfica ilustrativa que muestra las aplicaciones más relevantes en función de su categoría

¹⁸ Añadido en el diccionario de términos

C.1 Contraste con la competencia

Debido a que la revisión sistemática realizada contiene un gran número de surveys¹⁹ de fuentes válidas y recientes, creo que la carga de datos ha sido bastante completa y recoge la mayoría de las herramientas que utiliza el sector tecnológico aragonés. Esto se reafirma de esta manera, previo a las decisiones finales obtenidas por parte de los informes de “TECNARA Tools”, debido al gran elevado grado de afinidad con los propios reportes obtenidos del propio portal del CTA²⁰. Estos datos están basados en el análisis de sus cursos y las formaciones que estos imparten, ya que en dicho centro también se imparten cursos orientados a promover el desarrollo de nuevas tecnologías (*Portal CTA, 2020b*) y, por tanto, también este tipo de herramientas. Actualmente, dentro del dominio estudiado en el actual proyecto, ofertan dos cursos:

- **BIG DATA ARCHITECTURE WORKSHOP** (*Portal CTA, 2020a*): curso que se centra en los problemas de arquitectura de Big Data en general y los aplica al diseño de un sistema complejo y elaborado. Herramientas tratadas: Hadoop (HDFS), Hadoop(MapReduce), Spark, Hive/Impala, Metroz, Hbase, Kudu entre otras.
- **CIENTÍFICO DE DATOS BIG DATA CLOUDERA** (*Portal CTA, 2020c*): curso que ayuda a resolver problemas mediante las siguientes herramientas: R, Hadoop, Spark y las ventajas de Spark MLlib.

Posteriormente a la verificación de los cursos ofertados por el CTA, se podrá verificar si la carga de datos es adecuada, si así lo indican los informes finales generados por “TECNARA Tools”. Por otra parte, el análisis de la competencial se considera adecuado para este tipo de investigaciones, ya que permite una discusión entre expertos del sector.

¹⁹ Añadido en el diccionario de términos

²⁰ Añadido en el diccionario de términos

D Historias de usuario

En esta sección se presentan las diferentes historias de usuario, que representan las diferentes capacidades que debe cubrir la aplicación para salvaguardar los objetivos propuestos.

CLIENTES/COLABORADORES:

- Como usuario quiero poder buscar herramientas para verificar el uso que se hace de estas en el sector tecnológico de Aragón y si se adecúa a mis necesidades.
- Como usuario quiero poder realizar búsquedas por nombre, categoría o valoración para ampliar mi criterio de búsqueda según mis criterios predefinidos.
- Como usuario quiero poder valorar una aplicación para agregarle/quitarle valor.
- Como usuario quiero poder comentar una aplicación para poder expresar mis vivencias y/o experiencias con dicha aplicación.
- Como usuario quiero poder crear mi cuenta de usuario para poder utilizar dicho portal web.
- Como usuario quiero poder modificar mi cuenta de usuario.
- Como usuario quiero poder eliminar mi cuenta de usuario para poder dejar de utilizar dicha plataforma.
- Como usuario quiero poder responder a encuestas para expresar mi opinión sobre diferentes aspectos tratados

ADMINISTRADOR DEL SISTEMA:

- Como administrador quiero poder crear nuevas aplicaciones que no estaban contempladas previamente para añadirle valor a este portal.
- Como administrador quiero modificar una de las aplicaciones que se mostraban a los distintos usuarios para mantenerles siempre con una información adecuada.
- Como administrador quiero poder borrar comentarios de aplicaciones concretas para evitar aportaciones ofensivas.
- Como administrador quiero poder crear nuevas categorías para poder recoger nuevos criterios que hagan de contenedor a estas herramientas.
- Como administrador quiero poder modificar categorías para poder modificar la descripción, nombre o las aplicaciones que posee en su interior.
- Como administrador quiero poder borrar categorías para evitar que el portal se quede obsoleto o recoja aplicaciones que han evolucionado.
- Como administrador quiero poder borrar aplicaciones que se hayan quedado obsoletas para no crear confusiones a los usuarios.

- Como administrador quiero que todas las cuentas de usuario siempre estén vinculadas a empresas de Aragón.
- Como administrador quiero que la valoración sea un número del 1 al 5.
- Como administrador quiere poder crear encuestas para poder reafirmar los resultados obtenidos.
- Como administrador quiero poder añadir/eliminar dominios de correo reconocidos por el sistema para permitir/bloquear la creación de cuentas en representación de nuevas empresas.
- Como administrador quiero poder activar/desactivar encuestas para que estas sean realizadas en un periodo de tiempo determinado.
- Como administrador quiero poder añadir/eliminar usuarios a distintos grupos para orientar encuestas a esos grupos específicos.
- Como administrador quiero poder visualizar los resultados obtenidos tanto de aplicaciones como encuestas para poder tomar decisiones de relevancia.
- Como administrador quiero poder descargar informes con los resultados recopilados de las distintas empresas para verificar las distintas contribuciones en otros sistemas.

Finalmente, se aportaron portales web de similar comportamiento como ejemplo, aunque el demostrador a desarrollar se deseaba que estuviera centrado en el dominio específico del Big Data. Estos portales son:

- Alternativeto²¹
- Slant ²²
- G2²³

²¹ Alternativeto: <https://alternativeto.net>

²² Slant: <https://www.slant.co/tags/development>

²³ G2: <https://www.g2.com/software/programming>

E Requisitos de la aplicación

Código	Descripción
RFU1	La plataforma permite a un usuario crear una cuenta en el portal.
RFU2	La plataforma permitirá iniciar sesión a los usuarios con cuenta en el sistema
RFU3	El sistema permitirá ver los objetivos de TECNARA Tools
RFU4	La aplicación permite a un usuario que ha iniciado sesión consultar y modificar datos relevantes a esa cuenta.
RFU5	La aplicación permite a un usuario que ha iniciado sesión borrar su cuenta.
RFU6	La plataforma web permite realizar búsquedas de aplicaciones en función de la categoría que lo recoge
RFU7	La aplicación web permitirá al usuario buscar aplicaciones en función de la valoración de esta.
RFU8	La plataforma web permitirá al usuario buscar aplicaciones en función de su nombre alfabéticamente ascendente y descendente
RFU9	La aplicación web permitirá a un usuario valorar una aplicación agregando o quitando valor a estas.
RFU10	La plataforma web debe soportar que un usuario realice aportaciones sobre sus experiencias de una herramienta concreta, añadiendo comentarios.
RFU11	La aplicación web soportará que un usuario realice modificaciones o borrados sobre sus comentarios
RFU12	La aplicación permitirá a un usuario responder a encuestas activas
RFA1	La aplicación web permitirá al administrador crear una categoría que recoja una serie de aplicaciones
RFA2	La plataforma web permitirá al administrador modificar categorías existentes.
RFA3	La aplicación web permitirá al administrador borrar categorías existentes
RFA4	La plataforma web permitirá al administrador crear nuevas aplicaciones
RFA5	La plataforma web permitirá al administrador modificar y borrar aplicaciones.

- RFA6** La aplicación web permitirá mover una aplicación de categoría
- RFA7** La plataforma web permitirá consultar al administrador diferentes gráficas de los resultados recogidos
- RFA8** La plataforma web permitirá al administrador descargar informes sobre las distintas aplicaciones.
- RFA9** La aplicación debe permitir al administrador borrar comentarios inapropiados de la comunidad del portal.
- RFA10** El portal web permite añadir permisos de administrador a otros usuarios
- RFA11** El panel del administrador permite al administrador eliminar usuarios con el rol de usuario
- RFA12** El administrador podrá crear modificar y eliminar las encuestas que desee
- RFA13** El sistema debe permitir al administrador activar y desactivar una encuesta para delimitar el tiempo de respuesta
- RFA14** La aplicación permitirá al administrador la visualización de los resultados de las distintas encuestas
- RFA15** El administrador puede crear/modificar/eliminar grupos identificativos de personas que incluyan usuarios
- RFA16** La aplicación permite dirigir una encuesta a un dominio reducido de personas

Por otro lado, se presentan los requisitos no funcionales:

Código	Descripción
RNF1	La plataforma debe cumplir con la Ley Orgánica de Protección de Datos ²⁴
RNF2	La aplicación aportará una interfaz clara ²⁵ a la hora de la navegación.
RNF3	La aplicación debe poseer un diseño “Responsive” para garantizar su usabilidad en distintos dispositivos
RNF4	Los informes deben poderse descargar en formato CSV.
RNF5	Utilizar Docker para la implementación del sistema

²⁴ La aplicación debe proteger la privacidad de los usuarios.

²⁵ El usuario debe ser consciente de la información solicitada por parte de la web y entender el feedback que esta le devuelve. Además, debe comprender el flujo a llevar a cabo cada acción ofertada.

F Tecnologías utilizadas para la elaboración del portal

Respecto a las tecnologías para la implementación del portal, TECNARA dio total libertad. Es verdad, que existe gran variedad de tecnologías totalmente acoplables entre sí para este tipo de portales, pero se quería utilizar para la implementación del portal, una herramienta de Big Data de código libre de las analizadas previamente. Esta herramienta es Elasticsearch (herramienta clasificada previamente en la categoría de Search) un motor de búsqueda que permite desacoplar entre Front-end y Back-end las herramientas tradicionales CRUD.

En lo que respecta a la capa de presentación o front-end se eligió por comodidad, rapidez y eficiencia, Angular. Angular 9 es un framework que ya se había utilizado en asignaturas anteriores, tales como Arquitectura Software & Gestión de Proyecto Software.

En la capa dedicada a la lógica, se utiliza el framework de Node.js sustentado en la versión 13.12, debido a la gran cantidad de librerías que proporciona (incluyendo una librería para Elasticsearch para el motor, Echarts para gráficos o Crypto para el cifrado de datos) y por comodidad con el lenguaje de programación que utiliza. Para lidar con los datos se ha decidido utilizar TypeORM, la cual permite la comunicación con la BD de postgres, la cual es gestionada mediante la interfaz de usuario que proporciona la herramienta DBeaver en su versión 7.0.0.

Front-end	Back-end
<p>Angular + ngrx + nx</p> <p> RealWorld example app</p> <p> Star  312  Fork  80</p>	<p>NestJS + TypeORM</p> <p> RealWorld example app</p> <p> Star  784  Fork  165</p>

Figura 22 Combinación de tecnologías utilizadas

Para el tema de desplegar la web se ha utilizado Docker, el cual contiene las librerías necesarias para el despliegue en los servidores de TECNARA de forma rápida y eficaz. Entre los contenedores utilizados destacan:

- **Portainer** (*Portainer, 2018*): facilita la gestión de contenedores.
- **Dejavu** (*Appbaseio, 2019*): proporciona una interfaz gráfica para el motor de búsqueda Elasticsearch.
- **Postgres** (*Docker Official Images, 2019b*): contiene la BBDD
- **ElasticSearch** (*Docker Official Images, 2019a*): contiene el motor de búsqueda

G Diagramas de Caso de Uso

Casos de uso	Crear cuenta
Actores	Usuario no registrado
Referencias	RFU1
Definición	El caso de uso comienza cuando un usuario desea registrarse
Curso normal	<p>El usuario introduce su nombre</p> <p>El usuario introduce un email</p> <p>El usuario introduce una contraseña que contempla los campos especificados</p> <p>El usuario confirma de nuevo la contraseña, para evitar errores de confusión.</p>
Flujo alternativo 1	La cuenta es una cuenta institucional reconocida por la aplicación, en la cual se envía un correo al usuario para activar la cuenta.
Flujo alternativo 2	El dominio no es reconocido y se precisa de decisión por parte del administrador para aceptar o cancelar dicha petición.

Casos de uso	Mostrar Objetivos
Actores	Usuario no registrado / Usuario registrado / Administrador
Referencias	RFU3
Definición	El caso de uso comienza cuando un usuario desea conocer los objetivos de dicho portal
Curso normal	<p>El usuario accede a la pestaña de objetivos</p> <p>El usuario consulta los objetivos del portal web.</p>

Casos de uso	Buscar aplicación
Actores	Usuario registrado / Administrador
Referencias	RFU6 / RFU7 / RFU8
Definición	El caso de uso comienza cuando un usuario desea encontrar una aplicación con un determinado fin
Curso normal	<p>El usuario accede al portal e inicia sesión.</p> <p>El usuario filtra por la categoría a la que pertenece la aplicación deseada</p>
Flujo alternativo 1	El usuario filtra de manera alfabética (ascendente, descendente) o por valoración, entre otras.

Casos de uso	Valorar aplicación
Actores	Usuario registrado / Administrador
Referencias	RFU9
Definición	El caso de uso comienza cuando un usuario desea hacer una valoración sobre una determinada aplicación con el fin de agregar/quitar valor.
Curso normal	<p>El usuario accede al portal e inicia sesión</p> <p>El usuario hace uso sobre el caso de uso anterior, para buscar la aplicación deseada.</p> <p>El usuario utiliza la escala de estrellas para valorar la aplicación</p> <p>El usuario guarda dicha valoración.</p>

Casos de uso	Responder Encuesta
Actores	Usuario registrado / Administrador
Referencias	RFU12
Definición	El caso de uso comienza cuando un usuario desea responder una encuesta activada.
Curso normal	<p>El usuario selecciona el enlace recibido al email</p> <p>El usuario responde las preguntas de la encuesta</p>
Flujo alternativo 1	La encuesta ha sido desactivada o expirada y esta no permite ser respondida.

Casos de uso	Comentar aplicación
Actores	Usuario registrado / Administrador
Referencias	RFU10
Definición	El caso de uso comienza cuando un usuario desea hacer un comentario sobre una determinada aplicación con el fin de agregar/quitar valor.
Curso normal	<p>El usuario accede al portal e inicia sesión</p> <p>El usuario hace uso sobre el caso de uso anterior, para buscar la aplicación deseada.</p> <p>El usuario escribe el comentario que desea publicar</p> <p>El usuario confirma el comentario escrito</p>

Casos de uso	Crear Aplicación
Actores	Administrador
Referencias	RFA4
Definición	El caso de uso comienza cuando el administrador desea crear una aplicación
Curso normal	<p>El administrador accede al portal e inicia sesión</p> <p>El administrador hace uso del panel de administrador para la creación de una aplicación</p> <p>El administrador escribe el nombre de la aplicación</p> <p>El administrador selecciona la categoría a la que pertenece</p> <p>El administrador otorga de una descripción esa categoría</p> <p>El administrador selecciona una imagen sobre dicha aplicación</p> <p>El administrador confirma la creación de la aplicación</p>

Casos de uso	Modificar Aplicación
Actores	Administrador
Referencias	RFA5
Definición	El caso de uso comienza cuando el administrador desea modificar una aplicación existente
Curso normal	<p>El administrador accede al portal e inicia sesión</p> <p>El administrador hace uso del panel de administrador para la modificación de una aplicación</p> <p>El usuario selecciona la aplicación a modificar entre las aplicaciones existentes</p> <p>Elige el campo/campos a modificar (url, descripción, nombre...)</p> <p>Confirma su modificación</p>
Precondición	Debe existir al menos una aplicación

Casos de uso	Borrar Aplicación
Actores	Administrador
Referencias	RFA5
Definición	El caso de uso comienza cuando el administrador desea borrar una aplicación existente
Curso normal	<p>El administrador accede al portal e inicia sesión</p> <p>El administrador hace uso del panel de administrador para el borrado de aplicaciones</p> <p>El usuario selecciona la aplicación que desea borrar</p>
Precondición	Debe existir al menos una aplicación

Casos de uso	Crear Categoría
Actores	Administrador
Referencias	RFA1
Definición	El caso de uso comienza cuando el administrador desea crear una categoría que recoja una serie de aplicaciones.
Curso normal	<p>El administrador accede al portal e inicia sesión</p> <p>El administrador hace uso del panel de administrador para la creación de la categoría</p> <p>El usuario escribe el nombre de la categoría</p> <p>El usuario otorga de una descripción esa categoría</p>

Casos de uso	Modificar Categoría
Actores	Administrador
Referencias	RFA2
Definición	El caso de uso comienza cuando el administrador desea modificar una categoría de las existentes
Curso normal	<p>El administrador accede al portal e inicia sesión</p> <p>El administrador hace uso del panel de administrador para la modificación de la categoría</p> <p>El usuario selecciona las categorías existentes</p> <p>Elige el campo/campos a modificar</p> <p>Confirma su modificación</p>
Precondición	Debe existir al menos una categoría

Casos de uso	Borrar Categoría
Actores	Administrador
Referencias	RFA3
Definición	El caso de uso comienza cuando el administrador desea borrar una categoría que recoja una serie de aplicaciones
Curso normal	<p>El administrador accede al portal e inicia sesión</p> <p>El administrador hace uso del panel de administrador para el borrado de categorías</p> <p>El usuario selecciona la categoría que desea borrar.</p> <p>Confirma su modificación</p>
Precondición	Debe existir al menos una categoría

Casos de uso	Descargar Informes
Actores	Administrador
Referencias	RFA8
Definición	El caso de uso comienza cuando el administrador desea obtener informes que recojan las contribuciones de las distintas empresas
Curso normal	<p>El administrador accede al portal e inicia sesión</p> <p>El administrador hace uso de la pestaña de estadísticas</p> <p>El administrador selecciona descargar informes.</p>

Casos de uso	Crear Encuesta
Actores	Administrador
Referencias	RFA12
Definición	El caso de uso comienza cuando el administrador desea crear una encuesta
Curso normal	<p>El administrador accede al portal e inicia sesión</p> <p>El administrador hace uso del panel de encuestas para la creación de estas</p> <p>El administrador selecciona el nombre que desea para la encuesta.</p> <p>El administrador selecciona el número de preguntas que desea realizar.</p> <p>El administrador escribe las preguntas y selecciona crear encuesta.</p>

Casos de uso	Activar Encuesta
Actores	Administrador
Referencias	RFA13
Definición	El caso de uso comienza cuando el administrador desea activar una encuesta para dar comienzo a esta y orientarla a un grupo de usuarios.
Curso normal	<p>El administrador accede al portal e inicia sesión</p> <p>El administrador hace uso del panel de encuestas para la activación de esta</p> <p>El administrador selecciona la encuesta que desea activar.</p> <p>El administrador selecciona al grupo que desea encuestar</p> <p>El administrador selecciona activar.</p>

Casos de uso	Desactivar Encuesta
Actores	Administrador
Referencias	RFA13
Definición	El caso de uso comienza cuando el administrador desea desactivar una encuesta para dar fin a la recogida de respuestas
Curso normal	<p>El administrador accede al portal e inicia sesión</p> <p>El administrador hace uso del panel de encuestas para la desactivación de esta</p> <p>El administrador selecciona la encuesta que desea desactivar, de esta manera ninguna encuesta podrá someter ninguna respuesta más.</p>

H Vista de módulos

El estilo que se ha utilizado y respetado durante todo el ciclo de vida del proyecto ha sido un estilo basado en capas. El estilo detallado hablado realiza como una primera división en la que destacan las siguientes capas:

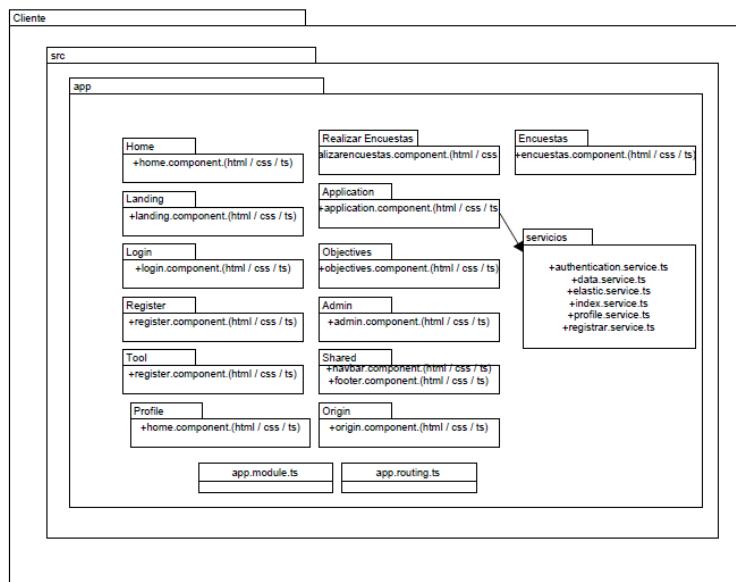


Figura 23 Diagrama de módulos: Cliente

- **Capa cliente:** es la capa de presentación al usuario, cuya principal funcionalidad es presentar una interfaz responsive, clara y cómoda para todo tipo de usuario que utilice el portal. Esta capa no solo presentará resultados, sino que es capaz de recoger las diferentes acciones llevadas a cabo por el usuario y conducirlas al destino adecuado.
 - **Servicios:** se definiría como el módulo que solicita la acción requerida al back end o al motor de búsqueda para funciones de consulta de datos en su dado caso.
 - **authentication.service.ts:** módulo que recoge todos los servicios que conciernen en la autenticación de usuarios, entre la que destaca el inicio de sesión (almacenando jwt) y logout.
 - **data.service.ts:** módulo que recoge todos los servicios referentes a datos de aplicaciones. Entre los que destacan, comentar, valorar, crear aplicaciones, realizar encuestas entre otras.
 - **elastic.service.ts:** que recoge todos los servicios referentes a búsquedas en el motor de búsqueda de Elasticsearch. Entre sus principales funciones son búsquedas con una serie de criterios como valoración, alfabéticamente o detalles de una aplicación/aplicaciones.

- **index.service.ts:** módulo que recoge todos los servicios referentes a los índices que trabajan como contenedores de las diferentes aplicaciones. Entre sus servicios destacan las operaciones CRUD de categorías.
- **profile.service.ts:** módulo que recoge los servicios referentes al perfil de un usuario como la modificación de credenciales, obtención de alguno de sus detalles personales o el borrado de una cuenta.
- **registrar.service.ts:** módulo que recoge los servicios referentes al registro de un usuario.

Dentro de la capa del cliente también se ha realizado una organización en la cual se contemplan vistas y componentes en su conjunto siempre en función de su objetivo. Todos actúan de igual manera, explicamos un ejemplo a continuación:

- **admin:** módulo que está formado por el fichero que se presenta al cliente (*.html) y el fichero que detalla el estilo de dicha capa de presentación (*.scss). Además, todos estos módulos también recogen un componente el cual es el responsable de llevar la lógica de la presentación e invocar al servicio correspondiente (*.component.ts).

Otros detalle destacables en la capa cliente es el módulo interceptor con 3 clases destacables: “jwt.interceptor.ts” el cual trabaja mediante eventos y en caso de realizar una llamada al backend estando logueado y con un JWT válido (no expirado) lo añade a la cabecera de las distintas peticiones, “error.interceptor.ts” clase que intercepta mediante eventos las respuestas del backend y en caso de producirse un código 401, conocido como Unauthorized realiza la salida de sesión y lo reconduce a la pestaña del login y finalmente “auth.guard.ts” clase que protege las rutas y solo les permite acceder a ellas si poseen sesión activa.

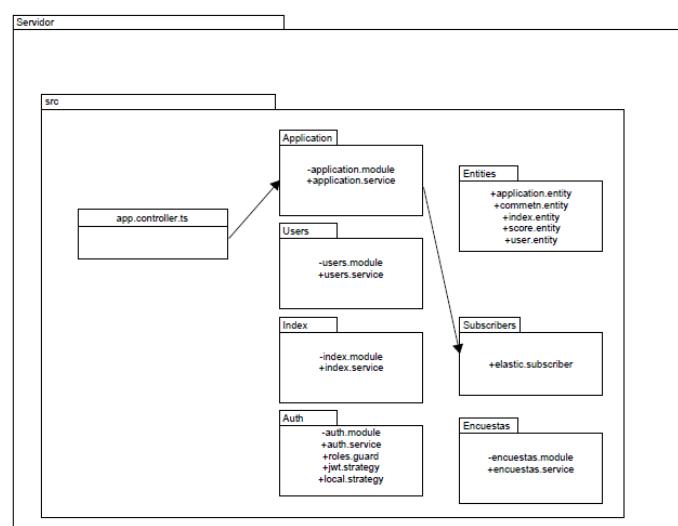


Figura 24 Diagrama de módulos: Servidor

- **Capa servidor:** es la capa de lógica de la aplicación. Encargada de realizar las peticiones solicitadas por los stakeholders y con la función característica de mantener en armonía la Base de Datos y el motor de búsqueda.
 - **Entities:** se definiría como el módulo que define y estructura el modelo de datos que va a soportar la aplicación. En este caso, se trabaja con las entidades Aplicación, Comentario, Categorías, Puntuaciones y Usuarios. Dentro de cada una de estas entidades se definen sus, campos, claves, valores por defectos y relaciones entre ellas (esto será explicado en capítulos posteriores).
 - **Subscriber:** se categoriza por ser el modelo que trabaja con la librería de ElasticSearch. Este módulo se suscribe mediante triggers a los distintos eventos que realiza el servidor hacia la BBDD y de esta forma se consigue realizarlos de manera secuencial en esta herramienta de Big Data.

De igual manera que se ha realizado en la capa cliente, se aplicó la misma forma de realización de módulos en función de su objetivo, esta vez dichos módulos recogen el módulo (ej.: application.module) encargado de proporcionar librerías y los componentes necesarios al servicio. Por otro lado, el servicio (ej: application.service) encargado de realizar las correspondientes modificaciones en la Base de Datos. El encargado de asociar las peticiones entrantes con el componente adecuado va a ser la clase “app.controller.ts” el cual detalla la guarda, rol, protocolo y parámetros requeridos para habilitar la funcionalidad requerida y que esta sea abordada por parte del servidor.

- **Capa BD:** capa más inferior de todo el sistema o portal. Esta capa proporciona y facilita la gestión de los datos para el funcionamiento correcto del Sistema. Es decir, en este sistema se producen la manipulación de los datos mediante operaciones de CRUD.
- **Capa ElasticSearch:** este módulo actuaría de igual manera que la BD cuando es actualizado por la capa servidor, pero solapa y modifica su manera de actuar de cara a la capa cliente. Esto se debe a que en este segundo caso el motor actúa ofreciendo una serie de APIS, es decir actuando como la capa servidor y expone una serie de documentos, es decir emulando el comportamiento de la BD.

I Diagrama de despliegue

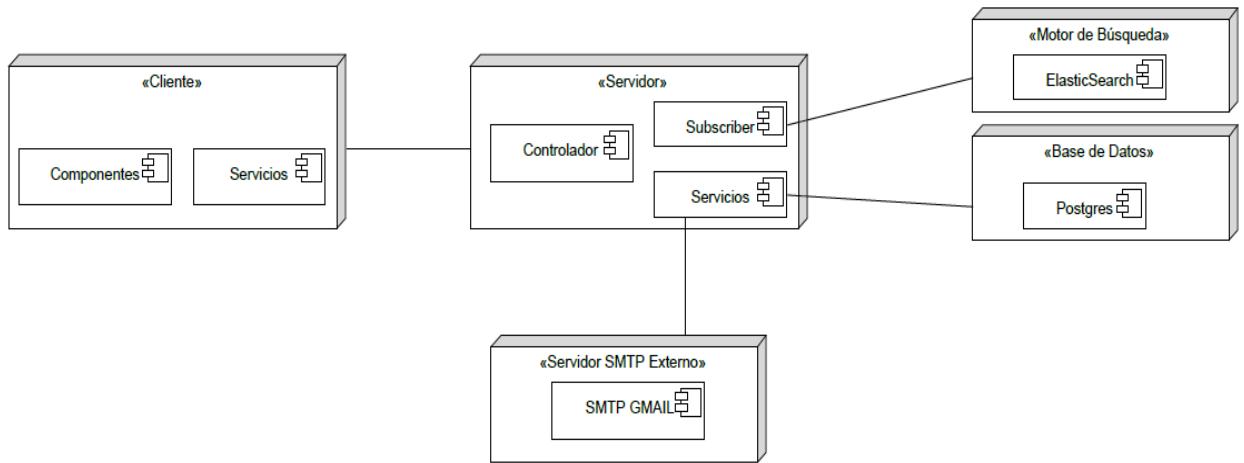


Figura 25 Diagrama de despliegue

- **Nodos:**

- Cliente: este nodo es el encargado de interaccionar con el usuario y traducir estas acciones para que el servidor sea capaz de llevar a cabo las consultas asociadas. Este a su vez está formado por una serie de componentes tales como: login, home, encuestas... cada uno de los componentes anteriores contiene su vista (*.html, *.css) y el comportamiento de esta (*.ts), el cual invocará a los servicios que harán uso de los distintos puertos creados. Estos puertos son Servidor y ElasticSearch. El primero le permite realizar operaciones de escritura, actualización y borrado y recibir realimentación por ello. Por otro lado, el puerto ElasticSearch encargado de las operaciones de lectura y obtener los datos asociados.
- Servidor: nodo cuya funcionalidad reside en la lógica del portal, el cual está formado por un controlador el cual redirige las llamadas a cada uno de los servicios implementados y se apoya en entidades para el acceso correcto a la capa de datos. Por lo que, esta capa gestiona las peticiones del cliente, realiza las iteraciones con la base de datos, sincroniza mediante hooks²⁶ tanto la base de datos, como el motor de búsqueda y da realimentación al cliente y, por consiguiente, al usuario. Para realizar el cometido de la sincronización posee dos puertos uno a cada componente de los que se procede a explicar a continuación.
- SMTP: servidor externo de Gmail el cual ha sido utilizado para enviar correos electrónicos a los distintos usuarios, que hacen uso del portal. Este ha sido elegido, por tiempo ya que de este modo no se configura nuestro propio servidor SMTP y

²⁶ Añadido a diccionario de términos

eficacia ya que disminuye la probabilidad de estar en una lista negra, siendo nuestros emails etiquetados como SPAM.

- BBDD: se encarga del almacenamiento de los datos del sistema. Este almacena y proporciona una API para la manipulación de estos desde el punto de vista del servidor. Este componente posee un único puerto, el cual le permite comunicarse con el servidor.
- ElasticSearch: componente que posee en armonía los datos contrastados con la Base de Datos. Este posee una serie de APIS integradas que oferta para su consumo al cliente y para su sincronización al servidor.

- **Conectores:**

- Acceso a BD: conector de tipo acceso a datos. Este permite la comunicación entre Servidor y la base de datos PostgreSQL. De esta forma, este conector permite que el servidor ejecute una serie de consultas o peticiones y le sean respondidas de manera adecuada.
- HttpRequest: conector de tipo *http*, que conduce las peticiones de manera asíncrona entre el componente cliente y su servidor. Esta relación no es lineal ya que un servidor puede atender a varios clientes, pero ese cliente solo puede consultar a ese servidor.
- Acceso a motor de búsqueda: conector que permite la consecución de los dos anteriores. Tanto el conector establecido entre cliente y motor, como el conector entre servidor y motor trabajan invocando los servicios ofertados por la API proporcionada por ElasticSearch, permitiendo ejecutar una serie de consultas y peticiones.
- Acceso a servidor SMTP: este conector requiere seguridad en la capa de transporte, por lo que se utiliza TLS. El puerto que permite la conexión entre ambos componentes es el 587 y requiere una cuenta de Gmail para el envío de estos.

J Modelado de datos

Entidad usuario:

La entidad Usuario es aquella que va a representar a todos los usuarios de la aplicación. Entre sus campos se encuentran nombre, email, password (la cual estará cifrada) y un rol. El campo más peculiar de dicha entidad es el campo Rol, el cual estará por defecto para los nuevos registros predefinido como “user” excepto las cuentas que destinamos para administración las cuales van definidas como “admin”.

En cuanto a relaciones, destaca que un usuario puede realizar varias valoraciones y varios comentarios, puede tener uno o más tags.

Entidad aplicación:

La entidad Aplicación es aquella que va a representar a todos las herramientas de Big Data OS (open source) obtenidas en el análisis previo. Estas aplicaciones se van a ver descritas por los campos: name, url, description, nota_media e index_id. Estos dos últimos serían los campos más destacables. El primero, nota_media va a ser el promedio de las valoraciones que realicen los distintos usuarios sobre una determinada aplicación y el segundo, index_id va a ser el campo que describe a qué categoría pertenece esa aplicación en el motor de búsqueda de ElasticSearch (es decir, Framework, Streaming...).

En cuanto a relaciones, destaca que una aplicación puede tener varias puntuaciones y varios comentarios. Además, que una aplicaciones pueden estar en más de un índice.

Entidad valoración/respuesta:

La entidad Valoración recogerá cada una de las puntuaciones realizadas por un usuario sobre una aplicación concreta. Esta tendrá una serie de restricciones como que un usuario no puede votar más de una vez sobre una misma aplicación. Esta está formada por el id del usuario y el id de la aplicación para limitar dicha funcionalidad.

En cuanto a sus dos relaciones con usuario y aplicación van a ser many to one lo que quiere decir que muchas valoraciones pueden haber sido realizadas por un usuario y muchas valoraciones pueden haber sido realizadas a una aplicación.

La entidad respuesta trabajará de igual manera que la entidad valoración ya que son homólogas. Valoración actuará para la funcionalidad de las aplicaciones y respuesta para la funcionalidad de las encuestas.

Entidad comentario:

La entidad Comentario trabaja exactamente igual que la entidad Valoración adecuada a sus campos descriptivos, excepto que comentarios, solo será editable durante las primeras 24 horas a su inserción.

Entidad índice:

La entidad índice es aquella que representa los índices que Elasticsearch utiliza para su motor de búsqueda. Esta se ha implementado debido a dos cosas:

- Los Id no permiten espacios, ni mayúsculas y de esta forma se puede utilizar en el front end por ejemplo Data Access en vez de “dataaccess”, que es como se llama verdaderamente el índice en el motor.
- De cara a tener sincronizada la BBDD <-> elastic ya que como se gestiona con triggers los eventos de creación de aplicaciones y categorías se puede diferenciar la creación de un índice (o categoría) al _doc que lleva dentro que es la aplicación.

La única relación que posee es con aplicación que una categoría puede tener varias aplicaciones

Entidad dominio:

Esta entidad es la encargada de la gestión de los dominios conocidos que permiten que se complete un registro sin acción por parte del administrador. Es decir, dominios tales como: “unizar.es”, “opel-vauxhall.es”, “everis.es” entre otras.

Entidad solicitud:

Solicitud, es el espacio de datos que almacena a un usuario en el supuesto escenario que la aplicación no reconozca su dominio como una empresa y, por tanto, está pendiente de ser aceptado/cancelado por parte del administrador. En caso de que el administrador acepte la petición, ya que reconoce la empresa de este, el usuario pasará a ser guardado en la entidad usuario como activo y ya podrá hacer uso del portal.

Entidad encuesta:

Entidad dedicada a dar nombre a las diferentes encuestas y a detallar si esta es accesible para los usuarios o no. Además, estas encuestas recogen la participación de los usuarios.

Entidad pregunta:

Entidad en la cual se definen las preguntas que serán mostradas y respondidas por los usuarios. Esta tiene relación con las entidades encuesta y respuesta.

Entidad tag:

Entidad que establece grupos específicos dentro de la comunidad que representan a subconjuntos de usuario. Entre ellos destacan alumnos, profesores y administradores.

Entidad asociación:

Entidad que relaciona un usuario con uno o más tags. De esta manera, se pueden realizar encuestas u otras acciones orientadas a usuarios en referencia al grupo/grupos que pertenecen.

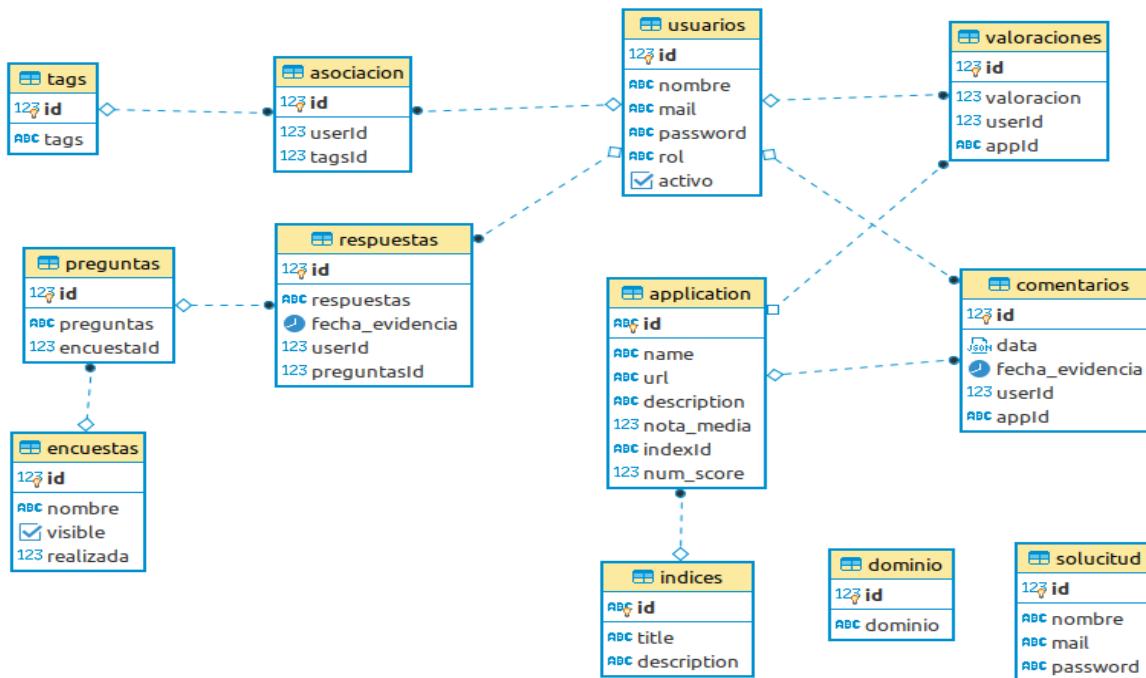


Figura 26 Diagrama Entidad-Relación

J.1 Sincronización entre ElasticSearch & BBDD

La sincronización entre Elastic & la BBDD es gestionado por el backend en Node. Este framework nos proporciona la librería “@elastic/elasticsearch”, para facilitarnos dicha sincronización y la cual utilizamos en su versión 7.6.1.

La sincronización se produce mediante la clase “elastic.subscriber”, la cual trabaja mediante suscripción de eventos. En función del comportamiento deseado utilizaremos dichos eventos para actualizar el motor de búsqueda. Se ha diferenciado cuando se crea/modifica/borra una categoría a cuando se crea/modifica/borra una aplicación, esta diferencia se debe a que las categorías van a actuar como índices de elastic, por el contrario, las aplicaciones trabajarán como docs recogidos dentro de dichos índices.

```

    /**
     * Subscriber de las entidades Aplicación e Indice, que son las que hay que
     * mantener sincronizadas con elasticsearch
     */
    async beforeInsert(event: InsertEvent<any>) {
        const categoria = event.metadata.name
        if(categoria=='Indice'){
            await this.insertarIndice(event.entity.id)
        }else if(categoria=='Application'){
            const index = event.entity.index
            delete event.entity.index
            const result = await this.insertarApp(index,event.entity)
            const idApp :string = result.body._id
            event.entity.id=idApp
            event.entity.indexId=index
        }
    }
}

```

Figura 27 Fragmento de código que permite la sincronización entre el motor y la BD

Como se puede ver se realizan las distinciones nombradas anteriormente en el fragmento de código previo. En caso de ser un “index event” se crea un nuevo índice que representaría a una nueva categoría (Framework, Streaming, Security...).

Por otro lado, si es un evento relacionado con una aplicación primero borramos el evento del índice, con el fin de evitar guardar en el doc, el índice al que pertenece, debido a que no queremos guardar la categoría dentro del doc sino la categoría trabaja como contenedor de dichas aplicaciones. Finalmente, se introducen los campos en el motor, los cuales que serán introducidos a continuación en la BBDD. Las siguientes imágenes muestran un ejemplo:

ElasticSearch:

<input type="checkbox"/> (...) _id	<input type="radio"/> description	<input type="radio"/> name	<input type="radio"/> nota_media	# num_score	<input type="radio"/> url
01 <input type="checkbox"/> (...) zm-dpnEBLjhhwQoBN572	Apache Beam es un fram	Beam			https://image.slidesharecdn.com
02 <input type="checkbox"/> (...) zW-dpnEBLjhhwQoBN57q	Apache Storm es un marco de trabajo para el desarrollo de aplicaciones de streaming en tiempo real.	Storm	3	1	https://blog.desdelinux.net
03 <input type="checkbox"/> (...) 7G-dpnEBLjhhwQoBOZ4u	Apache Kafka es un proyecto de software de código abierto que implementa un sistema de publicación/suscripción de datos en tiempo real.	Kafka	5	5	https://secure.meetupstatic.com
04 <input type="checkbox"/> (...) 0G-dpnEBLjhhwQoBOJ4E	Apache Apex es una plataforma de desarrollo de aplicaciones de streaming en tiempo real.	Apex	3	1	https://www.adaltas.com

Figura 28 Muestra las aplicaciones existentes en ElasticSearch

BBDD:

abc_id	abc_name	abc_url	abc_description	123 nota_media	abc_indexId	123 num_score
0G-dpnEBLjhhwQoBOJ4E	Apex	https://www.adaltas.com/	Apache Apex es una plataforma de desarrollo de aplicaciones de streaming en tiempo real.	3	streaming	1
zW-dpnEBLjhhwQoBN57q	Storm	https://blog.desdelinux.net	Apache Storm es un marco de trabajo para el desarrollo de aplicaciones de streaming en tiempo real.	3	streaming	1
7G-dpnEBLjhhwQoBOZ4u	Kafka	https://secure.meetupstatic.com	Apache Kafka es un proyecto de software de código abierto que implementa un sistema de publicación/suscripción de datos en tiempo real.	5	streaming	5
zm-dpnEBLjhhwQoBN572	Beam	https://image.slidesharecdn.com	Apache Beam es un marco de trabajo para el desarrollo de aplicaciones de streaming en tiempo real.	[NULL]	streaming	[NULL]

Figura 29 Muestra las aplicaciones existentes en la Base De Datos

Por eso es importante que las operaciones del administrador sean controladas desde el front end, para evitar confusiones de direccionamiento y evitar actualizaciones inconsistentes (por ejemplo, actualizaciones directas al motor de búsqueda sin que la BBDD sea consciente de ello).

J.2 Búsquedas en ElasticSearch

Para verificar la consistencia de los datos y la realización de querys, se ha utilizado el contenedor Dejavu. Dejavu se podría definir como la parte de front end que complementa a Elasticsearch. Por eso aprovechamos su funcionalidad, para la realización de los métodos implementados en el front end, como búsquedas alfabéticas (asc / desc), por valoración, entre otras.

Figura 30 Muestra el generador de consultas proporcionado por Dejavu

La captura anterior, muestra que Dejavu proporciona la serie de campos que debemos configurar para hacer la llamada al motor de búsqueda y así conseguir la funcionalidad deseada. En esa misma captura, se puede apreciar que devuelve una lista, con el conjunto de aplicaciones ordenadas en función de su nota media. En este caso, proporciona primero la aplicación Kafka (con un 5 de nota media) y posteriormente la aplicación Apex (3 de nota media). Ambas aplicaciones dentro de la categoría de Streaming

```
"hits": [],
{
  "_index": "streaming",
  "_type": "_doc",
  "_id": "7G-dpnEBLjhhwQoB0Z4u",
  "score": null,
  "_source": {
    "name": "Kafka",
    "url": "https://secure.meetupstatic.com/photos/event/d/0/6/4/600_472013348.jpeg",
    "description": "Apache Kafka es un proyecto de intermediación de mensajes de código abierto desarrollado por LinkedIn y donado a la Apache Software Foundation.",
    "nota_media": 5,
    "num_score": 5
  },
  "sort": [
    "5"
  ]
},
{
  "_index": "streaming",
  "_type": "_doc",
  "_id": "0G-dpnEBLjhhwQoB0J4E",
  "score": null,
  "_source": {
    "name": "Apex",
    "url": "https://www.adaltas.com/static/dfbb7f51ae4e8b70b243eae039aad100/ccef1/apache-apex.png",
    "description": "Apache Apex es una plataforma nativa de YARN que unifica la transmisión y el procesamiento por lotes. Procesa grandes datos en movimiento.",
    "nota_media": 3,
    "num_score": 1
  },
  "sort": [
    "3"
  ]
}
```

Figura 31 Muestra los resultados devueltos por ElasticSearch

J.3 Protección de rutas

En esta aplicación, se han protegido algunas de las rutas del front end, para que no sean accesibles a usuarios sin un previo inicio de sesión. Las rutas que van a ser accesibles independientemente de si estás logueado o no van a ser las siguientes: “/home”, “/register”, “/objective”, “/origin” y “/login”.

```
{ path: 'home', component: HomeComponent },
{ path: 'user-profile', component: ProfileComponent, canActivate: [AuthGuard] },
{ path: 'register', component: SignupComponent },
{ path: 'objective', component: ObjectiveComponent },
{ path: 'landing', component: LandingComponent, canActivate: [AuthGuard] },
{ path: 'application/:id', component: ApplicationComponent, canActivate: [AuthGuard] },
{ path: 'tool/:category/:id', component: ToolComponent, canActivate: [AuthGuard] },
{ path: 'login', component: LoginComponent },
{ path: 'admin', component: AdminComponent, canActivate: [AuthGuard] },
{ path: 'estadisticas', component: EstadisticasComponent, canActivate: [AuthGuard] },
{ path: 'origin', component: OriginComponent },
{ path: '', redirectTo: 'home', pathMatch: 'full' }
```

Figura 32 Fragmento de código que protege las rutas de accesos sin sesión activa

El resto de las rutas estarán protegidas por el método canActivate() que pertenece a la clase AuthGuard. Dicho método devuelve un valor booleano en función de si existe un currentUser, es decir si existe alguna sesión activa.

```
canActivate(route: ActivatedRouteSnapshot, state: RouterStateSnapshot) {
  const currentUser = this.authenticationService.currentUserValue;
  if (currentUser) {
    // logged in so return true
    return true;
  }
  // not logged in so redirect to login page with the return url
  this.router.navigate(['login']);
  return false;
}
```

Figura 33 Fragmento de código que verifica la existencia de una sesión activa

J.4 Diferencias entre roles

Complementariamente a la protección de rutas en función de sesiones activas también se realiza una diferenciación de acceso en función del rol de usuario tanto en el front end como en el back end. El front end al iniciar una sesión válida siempre recibirá un JWT (Json Web Token) y un rol que acompañará al usuario en toda su experiencia a lo largo de nuestra web. Siempre y cuando no caduque, en caso de caducar redireccionará al login para volver a obtener dichos parámetros. El resultado obtenido puede ser de estas dos maneras en función del rol:

```
{
  "access_token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpZCI6MywidXNlc5hbWJ0iJBZG1pbmlzdHJhZG9yIiwibWFpbCI6ImFkbWluQGdtYWlsLmNvbSIsInJvbCI6ImFkbWluIiwiaWF0IjoxNTg3ODAwOTMxLCJleHAiOjE10Dc4MTQ1MzF9.cDRUwBkDW4CubjpfC7Rw3X2RnA1qZ01W9ebaNU5qdjs",
  "rol": "admin"
}
```

Figura 34 Respuesta del servidor para un acceso de sesión de un administrador

Recogiendo dichos campos se ha limitado alguna de las funcionalidades a usuarios tales como la recolección de estadísticas, descarga de informes u operaciones CRUD en torno a categorías o aplicaciones.

En la parte del back end se proporciona una serie de funciones invocables desde el front end, pero estas variarán en función del rol esperado. Un ejemplo de esto sería que las aplicaciones solo van a poder ser insertadas por un administrador y en lado contrapuesto van a poder ser comentadas por cualquier usuario.

Añadir App (accesible dependiente al rol):

```
/**  
 * @param app  
 */  
@Post('/add/app')  
@UseGuards(AuthGuard('jwt'))  
@Roles('admin')  
async addApp(@Body() app: Application): Promise<any> {  
    return await this.applicationService.addapplication(app);  
}
```

Figura 35 Fragmento de código que habilita la invocación de una funcionalidad sólo al administrador

Comentar App (accesible independientemente del rol):

```
/**  
 * @param app  
 * @param req  
 */  
@Post('/add/comment/:app')  
@UseGuards(AuthGuard('jwt'))  
async comentarApp(@Param('app') app: string, @Req() req, @Body('comment') comment: Comments): Promise<any> {  
    const user = req.user.id  
    return await this.applicationService.comentar(app, user, comment)  
}
```

Figura 36 Fragmento de código que habilita la invocación de una funcionalidad a todos los usuarios

La diferencia de estos es que la etiqueta `@Roles()`, verificará el rol que tiene dicha sesión o en su defecto dicho usuario, permitiendo su acceso si coincide con el pattern “admin”.

J.5 Descargar Informes

La funcionalidad de descargar informes CSV recogidos desde la BBDD, funciona de manera similar a cualquier procedimiento que finalmente es mostrado en el front end, es decir el método `getReport()` [utilizado para proporcionar informes] funciona de igual manera que el procedimiento `getTools()` [que devolvería todas las aplicaciones, en caso de no se utilizara ElasticSearch] a diferencia que este realiza una transformación para devolver como primera fila los campos a los que hace referencia para proporcionar una mayor claridad al informe

```

async getReport(): Promise<any>{
  const fields = ['id', 'name', 'url', 'description', 'nota_media', 'indexId', 'num_score'];
  const opts = { fields };
  const fields2 = ['id', 'title', 'description'];
  const opts2 = { fields2 };
  const myData = await this.appRepository.find();
  const myData2 = await this.indiceRepository.find();

  try {
    const parser = new Parser(opts);
    const csv = parser.parse(myData);
    const parser2 = new Parser(opts2);
    const csv2 = parser2.parse(myData2);
    return {"data":csv, "data2":csv2};
  } catch (err) {
    console.error(err);
  }
}

```

Figura 37 Fragmento de código que recopila los datos para el informe

Cuando dichos datos son recibidos por el front end, este espera dos tipos de información. File, el cual recogerá la información referente a las aplicaciones y file2, que contendrá la información referente a los índices. Dicha información genera un BLOB (formato indeterminado) y lo parsea a un formato CSV de descarga.

```

descargar() {
  console.log('lentro al controller')
  this.dataservice.descargarCSV().subscribe(
    response => {
      console.log(response.toString())
      let file = new Blob([response.data], { type: 'text/csv' });
      let file2 = new Blob([response.data2], { type: 'text/csv' });
      this.guardar(file,'aplicaciones.csv')
      this.guardar(file2,'indices.csv')
    },
  )
}

```

Figura 38 Fragmento de código que convierte los datos en formato CSV

Finalmente, se utiliza un “file saver” proporcionado por angular para la descarga de archivos. El método guardar (file, nombre), donde file es el fichero que se va a guardar y nombre, el string del fichero descargado.

```

async guardar(file,nombre) {
  await saveAs(file, nombre);
}

```

Figura 39 Fragmento de código que permite la descarga de los informes

J.6 Gráficas

Los datos de las gráficas van a ser recopilados de Elastic, debido al desacople que se ha realizado entre el front end y el back end (excepto para las funciones de las operaciones CRUD en los procedimientos esenciales). Para la parte visual, se ha procedido a utilizar la librería Echarts, proporcionada por el framework de Angular. Posteriormente a instalar dicha librería, se han implementado dos gráficas. La primera, recogerá las aplicaciones y la nota media de las valoraciones obtenidas. La segunda gráfica, recogerá las aplicaciones y la cantidad de votaciones recogidas. De esta manera se podrá contrastar información, ya que no es lo mismo un 5 de una votación que un 5 de 100 valoraciones recogidas.



```
1 this.chartOption = {  
2   xAxis: {  
3     type: 'category',  
4     data: this.nameAplication  
5   },  
6   yAxis: {  
7     type: 'value'  
8   },  
9   series: [{  
10     data: this.scoreAplication,  
11     type: 'bar'  
12   }]  
13 }  
  
1 this.chartOption2 = {  
2   xAxis: {  
3     type: 'category',  
4     data: this.nameAplication  
5   },  
6   yAxis: {  
7     allowDecimals: false,  
8     type: 'value'  
9   },  
10  series: [{  
11    data: vecesPuntuado,  
12    type: 'bar'  
13  }]  
14 }
```

Figura 40 Fragmento de código que configura las opciones de las gráficas

J.7 Seguridad

Se ha utilizado un protocolo de autenticación JWT (JSON Web Token). Este aporta seguridad para peticiones entre cliente y servidor. El funcionamiento de este mecanismo se basa en el transporte de un objeto JSON, el cual está encriptado. De manera que, este es generado por el servidor, cuando un usuario inicia una sesión válida. Las características que se han configurado en el JWT utilizado se centran la expiración de este token en el plazo de una hora. De manera que el cliente, tiene configurado un interceptor que verifica si recibe un error de expiración y lo redirecciona a hacer un nuevo inicio de sesión y, por tanto, conseguir un nuevo token.

J.8 Diagramas de secuencia

Registrar: diagrama de secuencia que describe la interacción entre los distintos componentes en el proceso de registro.

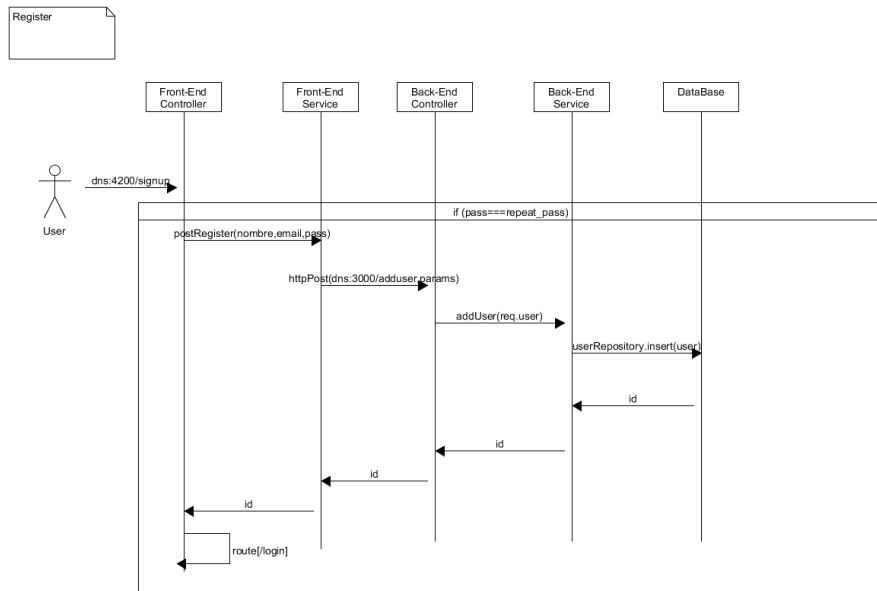


Figura 41 Diagrama de secuencia: Registro.

Login: diagrama de secuencia que describe la interacción entre los distintos componentes en el proceso de inicio de sesión.

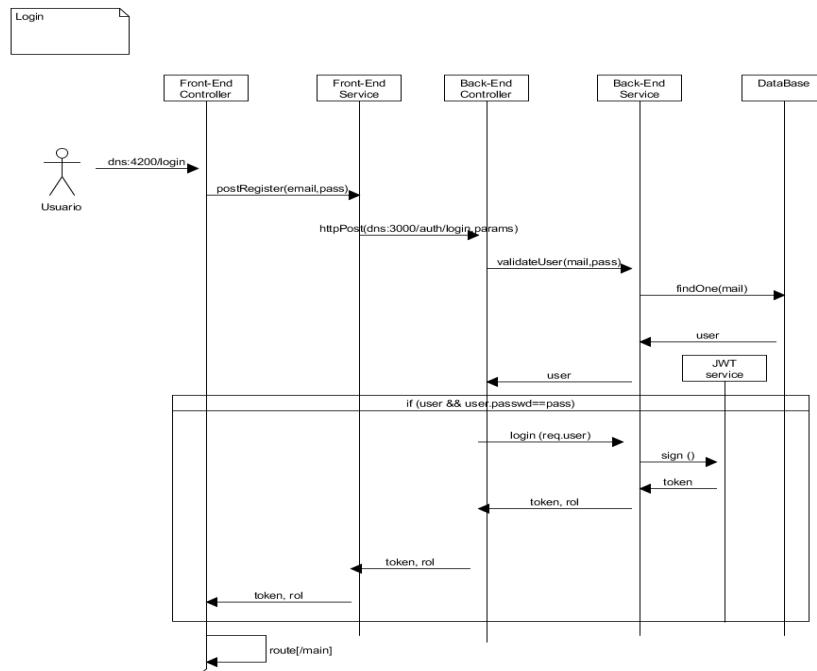


Figura 42 Diagrama de secuencia: Inicio de Sesión.

Profile: diagrama de secuencia que describe la interacción entre los distintos componentes al obtener los distintos datos de usuario.

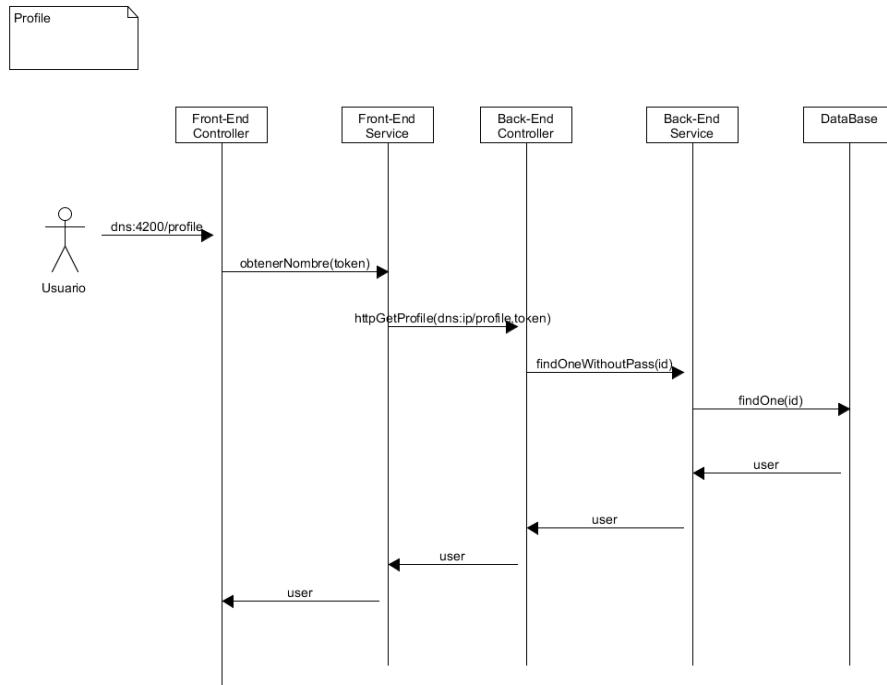


Figura 43 Diagrama de secuencia: Perfil de Usuario.

Actualizar contraseña: diagrama de secuencia que describe la interacción entre los distintos componentes en el proceso de actualización de contraseña.

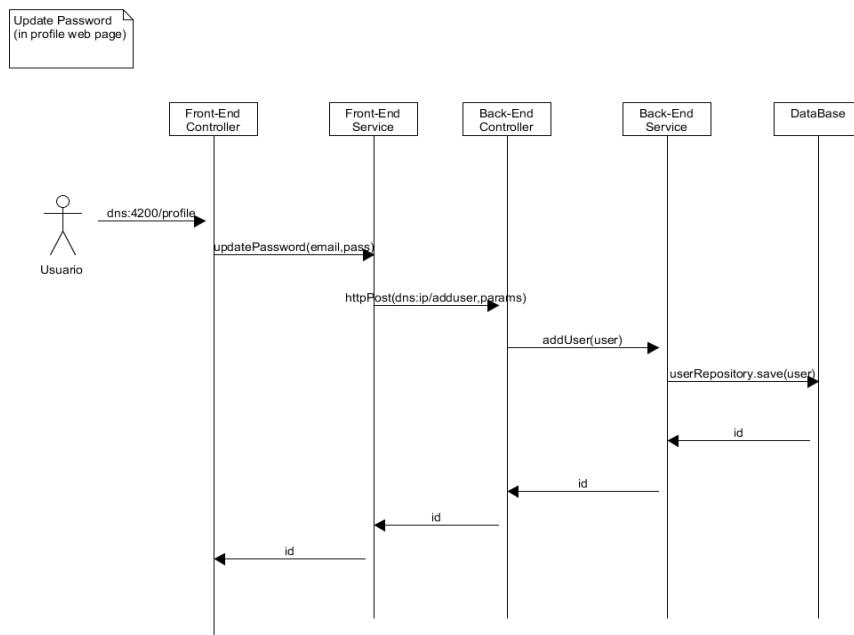


Figura 44 Diagrama de secuencia: Actualización de Credenciales.

Valorar: diagrama de secuencia que describe la interacción entre los distintos componentes en el proceso de valoración de una aplicación.

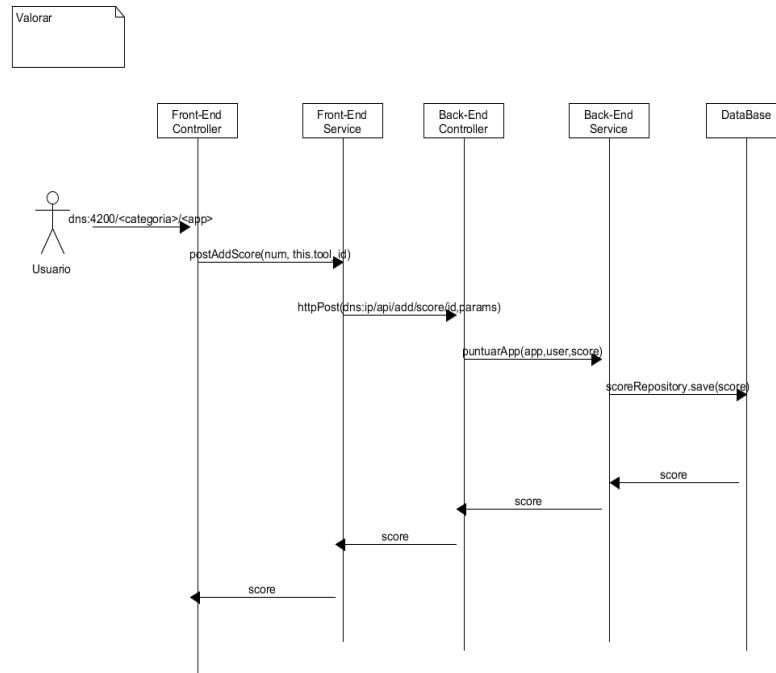


Figura 45 Diagrama de secuencia: Valorar una aplicación.

Comentar: diagrama de secuencia que describe la interacción entre los distintos componentes en el proceso de comentar una aplicación.

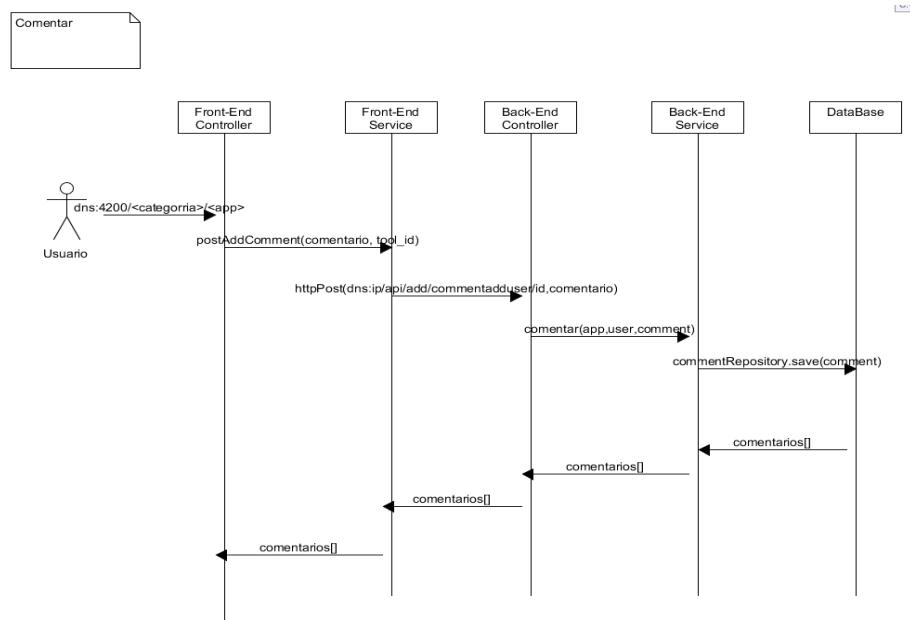


Figura 46 Diagrama de secuencia: Comentar una aplicación.

Crear categoría: diagrama de secuencia que describe la interacción entre los distintos componentes en el proceso de crear una nueva categoría.

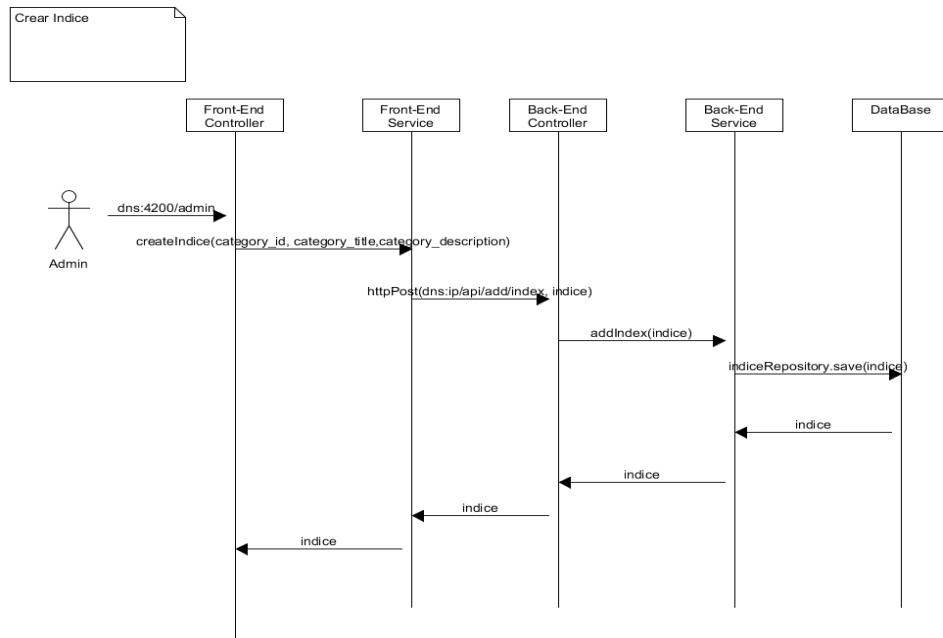


Figura 47 Diagrama de secuencia: Crear Categoría.

Modificar categoría: diagrama de secuencia que describe la interacción entre los distintos componentes en el proceso de modificar una nueva categoría.

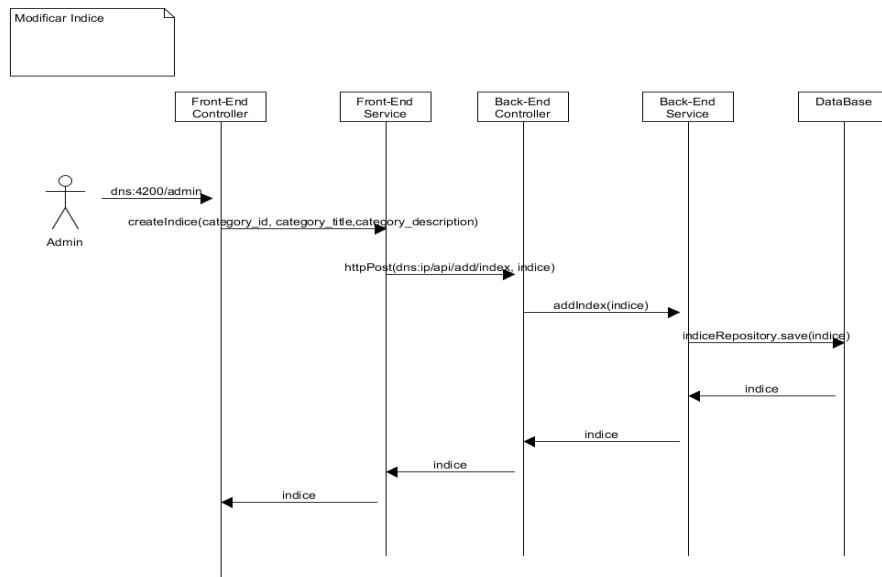


Figura 48 Diagrama de secuencia: Modificar Categoría.

Borrar categoría: diagrama de secuencia que describe la interacción entre los distintos componentes en el proceso de borrar una nueva categoría.

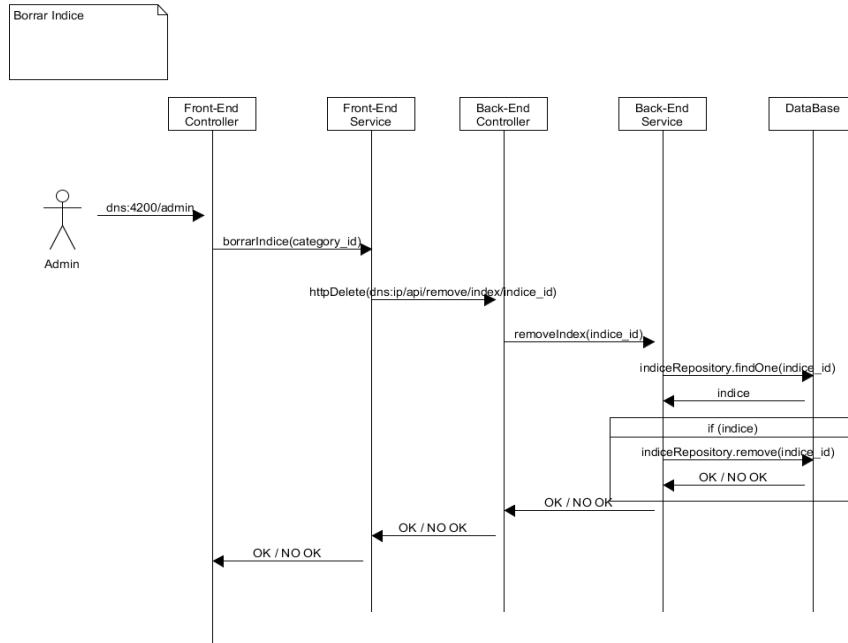


Figura 49 Diagrama de secuencia: Borrar Categoría.

Crear aplicación: diagrama de secuencia que describe la interacción entre los distintos componentes en el proceso de crear una nueva aplicación.

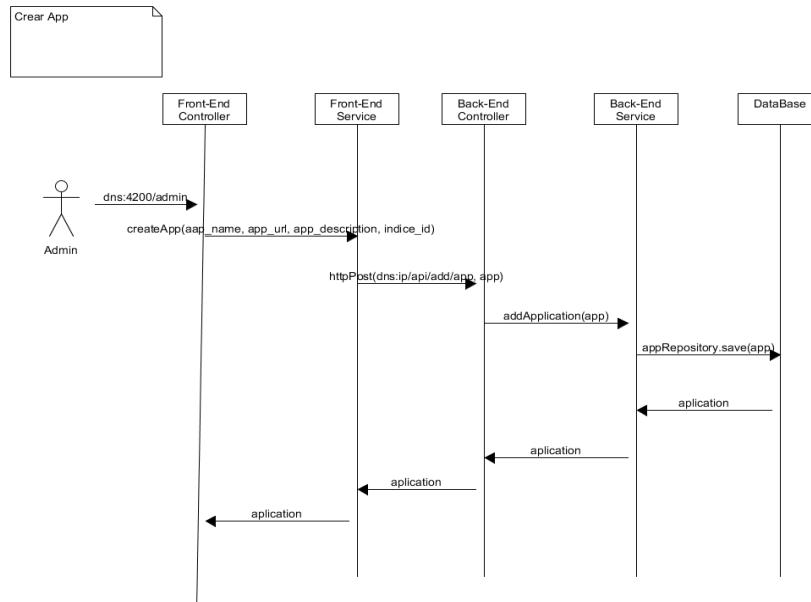


Figura 50 Diagrama de secuencia: Crear Aplicación.

Modificar aplicación: diagrama de secuencia que describe la interacción entre los distintos componentes en el proceso de modificar una aplicación.

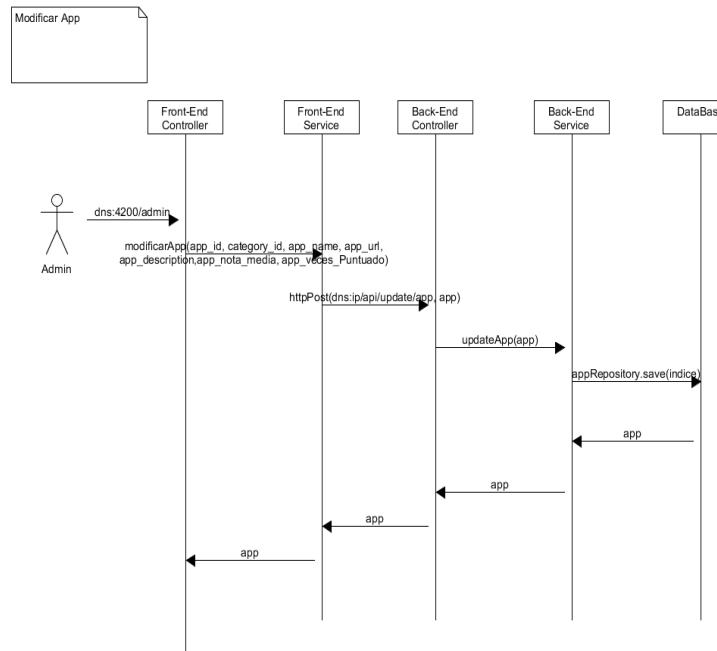


Figura 51 Diagrama de secuencia: Modificar Aplicación.

Borrar aplicación: diagrama de secuencia que describe la interacción entre los distintos componentes en el proceso de borrar una aplicación.

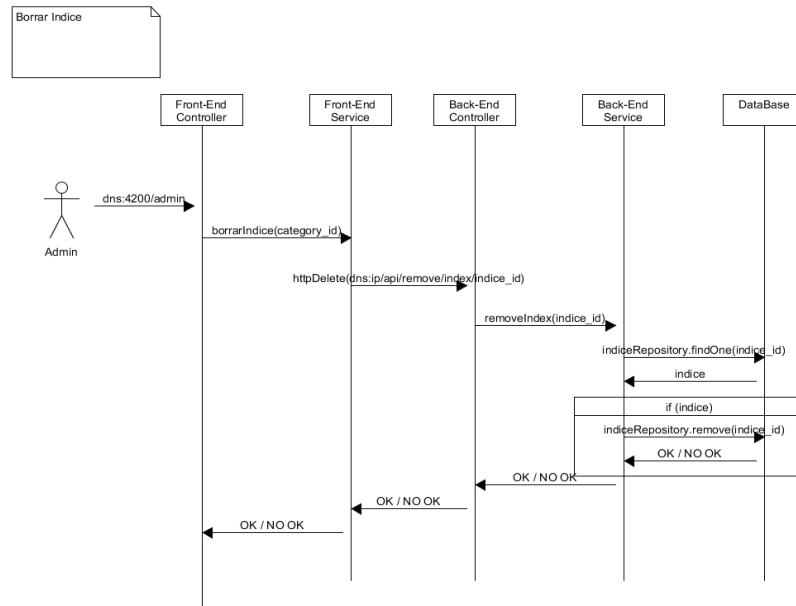


Figura 52 Diagrama de secuencia: Borrar Aplicación.

Borrar cuenta: diagrama de secuencia que describe la interacción entre los distintos componentes en el proceso de borrar una cuenta de usuario.

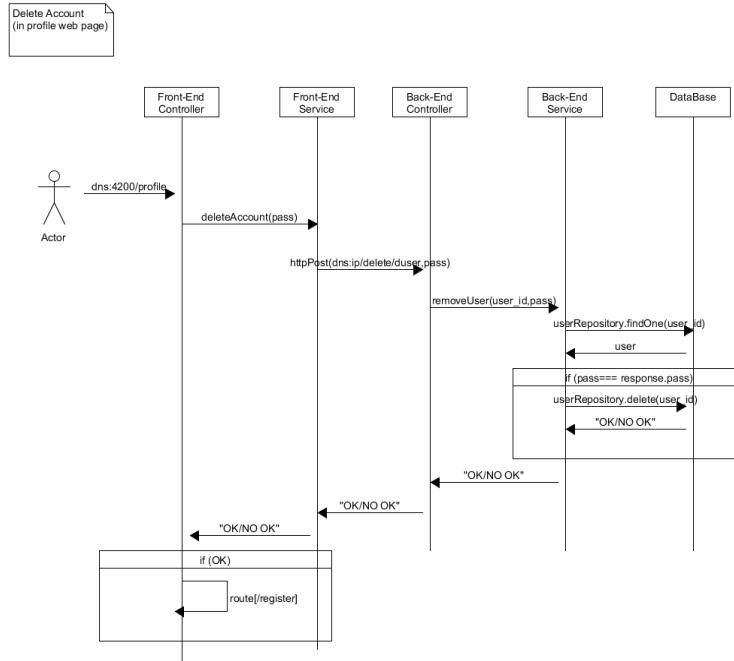


Figura 53 Diagrama de secuencia: Borrar Cuenta.

Obtener categorías: diagrama de secuencia que describe la interacción entre los distintos componentes en el proceso de devolver todas las categorías.

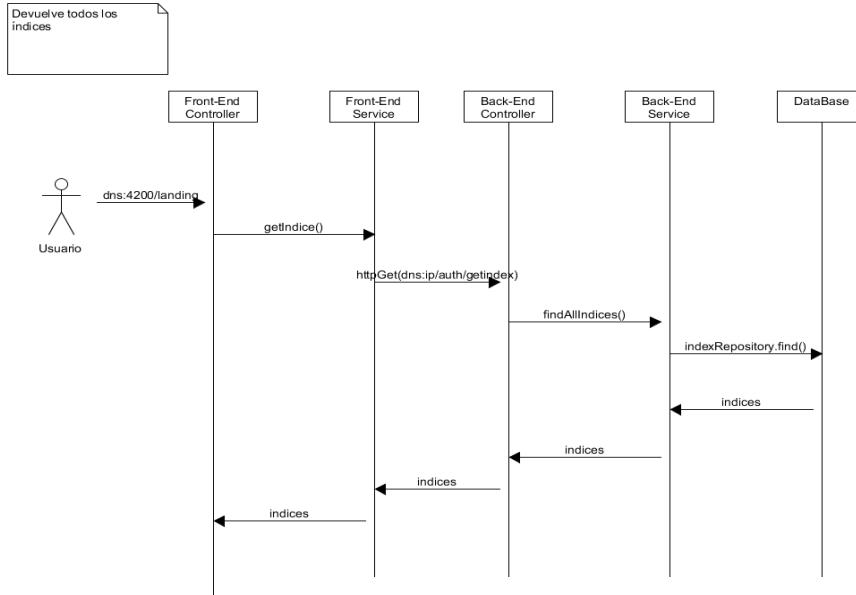


Figura 54 Diagrama de secuencia: Obtener Categorías

Obtener aplicaciones de una categoría (Elasticsearch): diagrama de secuencia que describe la interacción entre los distintos componentes en el proceso de devolver la aplicación de una categoría.

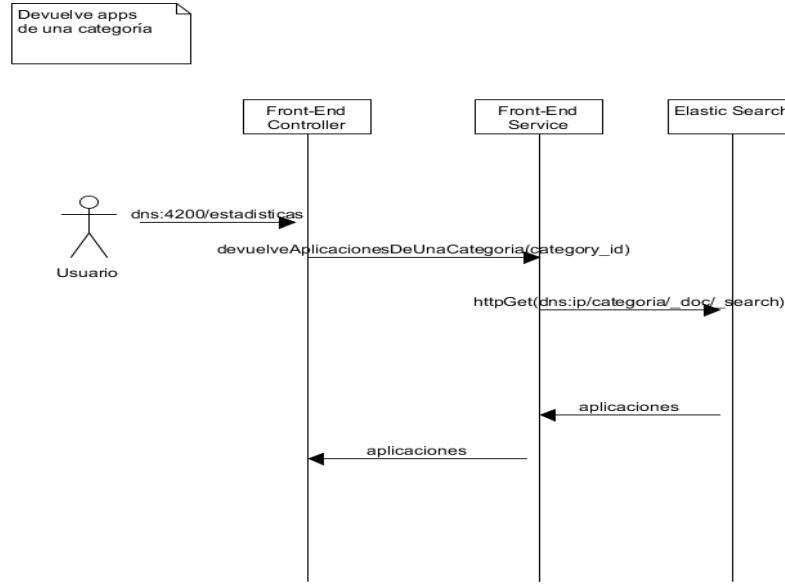


Figura 55 Diagrama de secuencia utilizando Elasticsearch: Obtener Aplicaciones de una categoría.

Devolver aplicaciones de una categoría (Back End): diagrama de secuencia que describe la interacción entre los distintos componentes en el proceso de devolver la aplicación de una categoría.

Este solo se utiliza para mostrar las diferencias con el diagrama anterior y mostrar las ventajas que Elasticsearch transmite a nuestra aplicación, pero no quiere decir que esté implementado de esta manera.

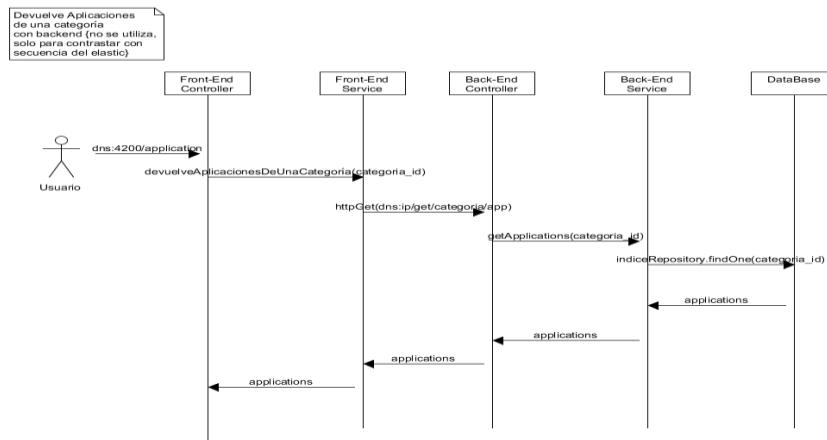


Figura 56 Diagrama de secuencia utilizando el Servidor: Obtener Aplicaciones de una categoría.

Obtener aplicaciones: diagrama de secuencia que describe la interacción entre los distintos componentes en el proceso de devolver las aplicaciones.

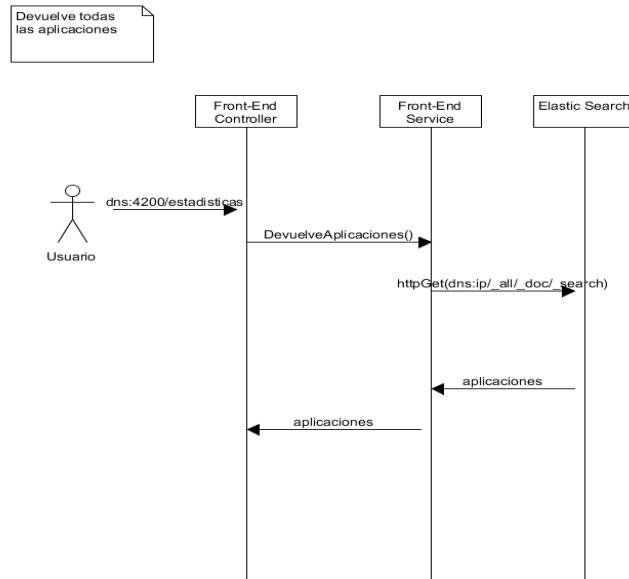


Figura 57 Diagrama de secuencia: Obtener todas las aplicaciones.

Descargar Informes: diagrama de secuencia que describe la interacción entre los distintos componentes en el proceso de descargar informes.

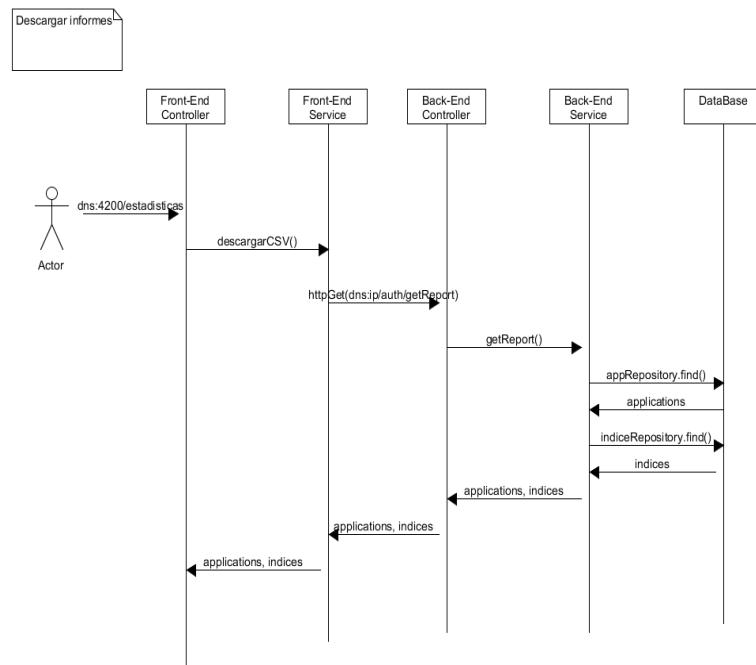


Figura 58 Diagrama de secuencia: Descargar Informes.

K Mapa de navegación

En lo que abarca a la sección relacionada con el mapa de navegación se ha realizado una representación de sus dos principales dispositivos. El primero, desde una web y un dispositivo de dimensión reducida (smartphone o similar) y desde un PC. Cabe destacar en ambos diagramas que los links se han representado con tres colores. Azul, enlaces que no hace falta tener una cuenta o haber iniciado sesión. Naranja, enlaces que es obligatorio tener una cuenta para utilizar las funcionalidades allí presentadas y rojo, acceso limitado sólo para las cuentas que poseen un rol de administrador.



Figura 59 Mapa de navegación para dispositivos móviles o similares

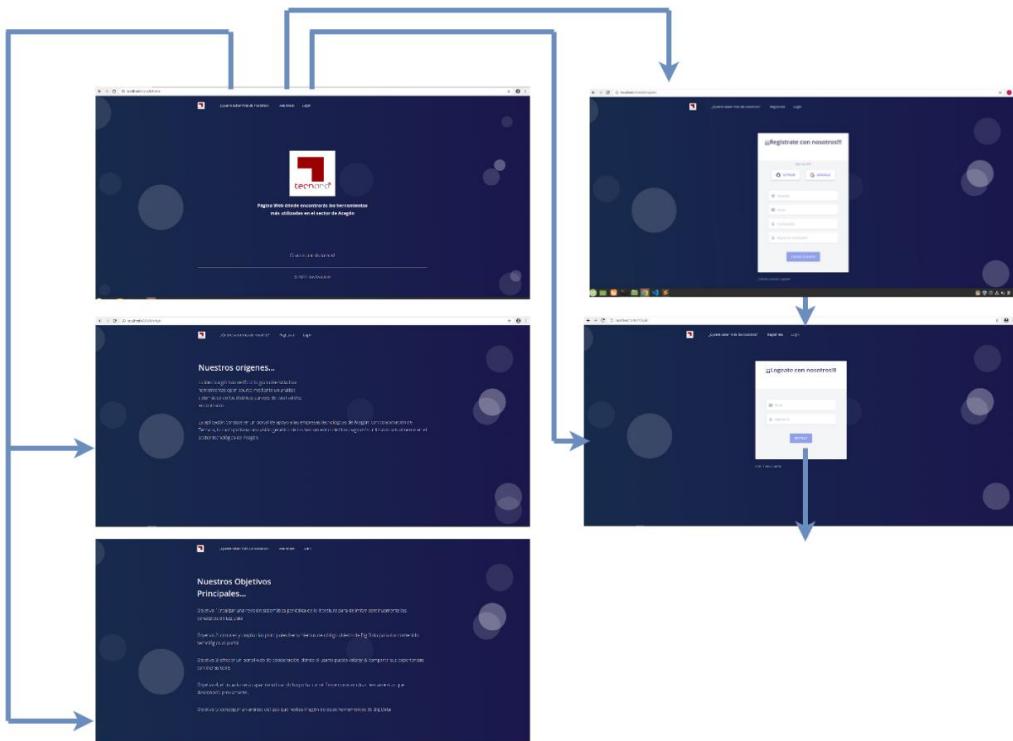


Figura 60 Mapa de navegación para ordenadores: Usuario no registrado

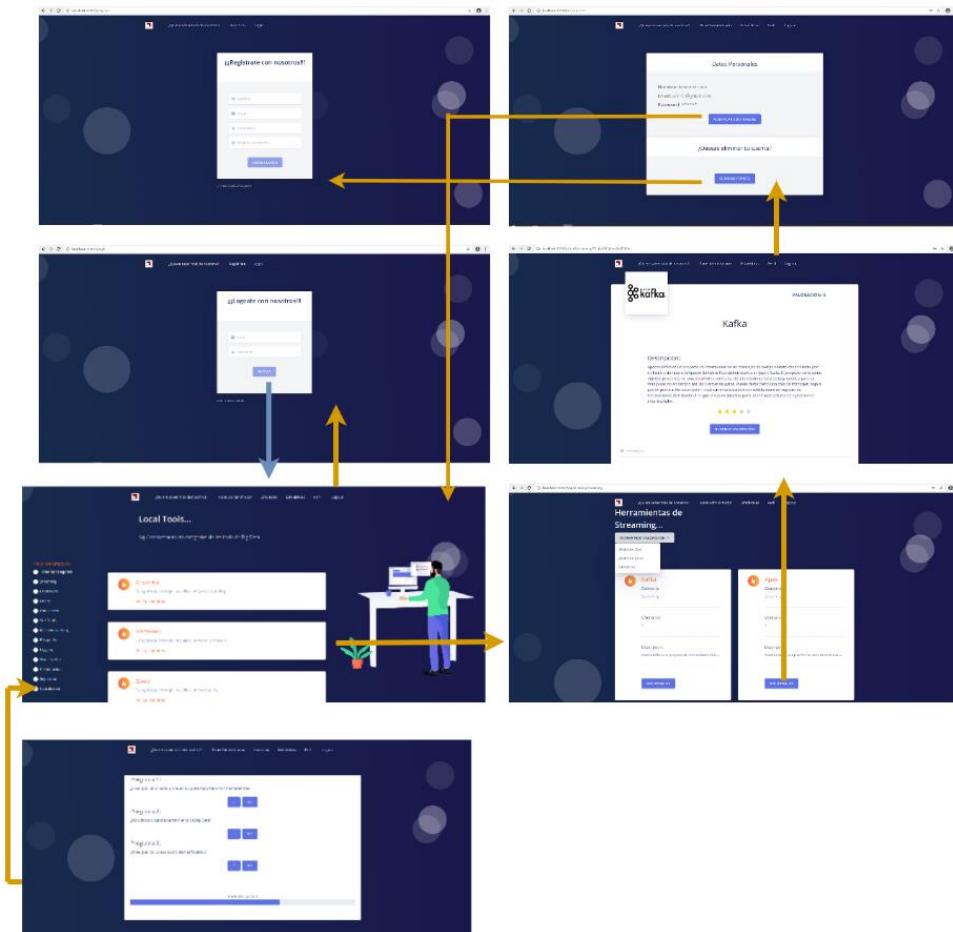


Figura 61 Mapa de navegación para ordenadores: Usuarios registrado

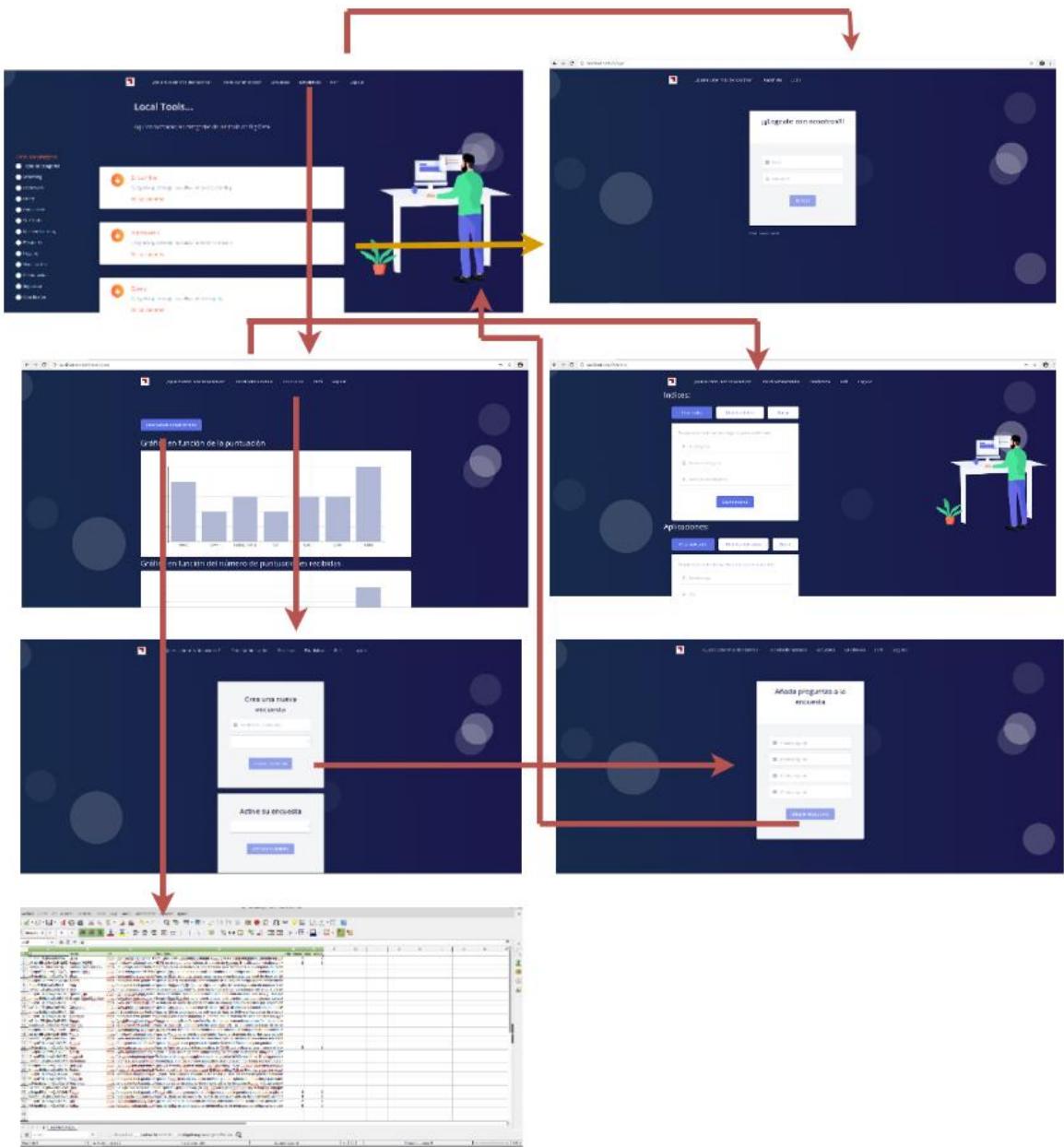


Figura 62 Mapa de navegación para ordenadores: Administrador

L Interfaces de usuario

Home:

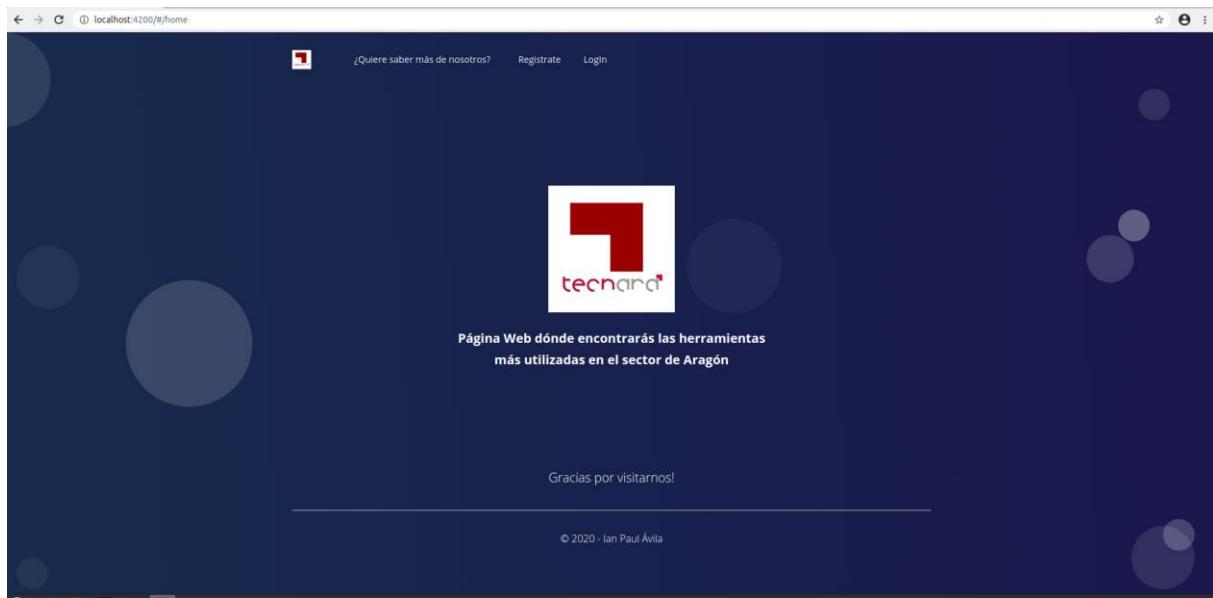


Figura 63 Pantalla: Home.

Origen:

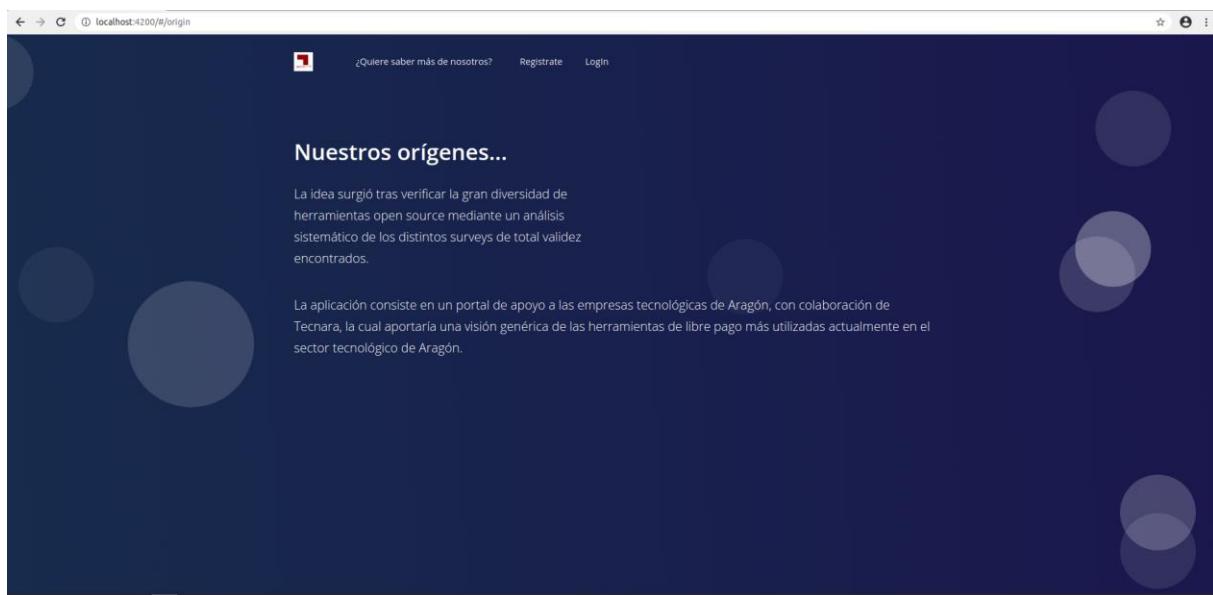


Figura 64 Pantalla: Origen.

Objetivos:

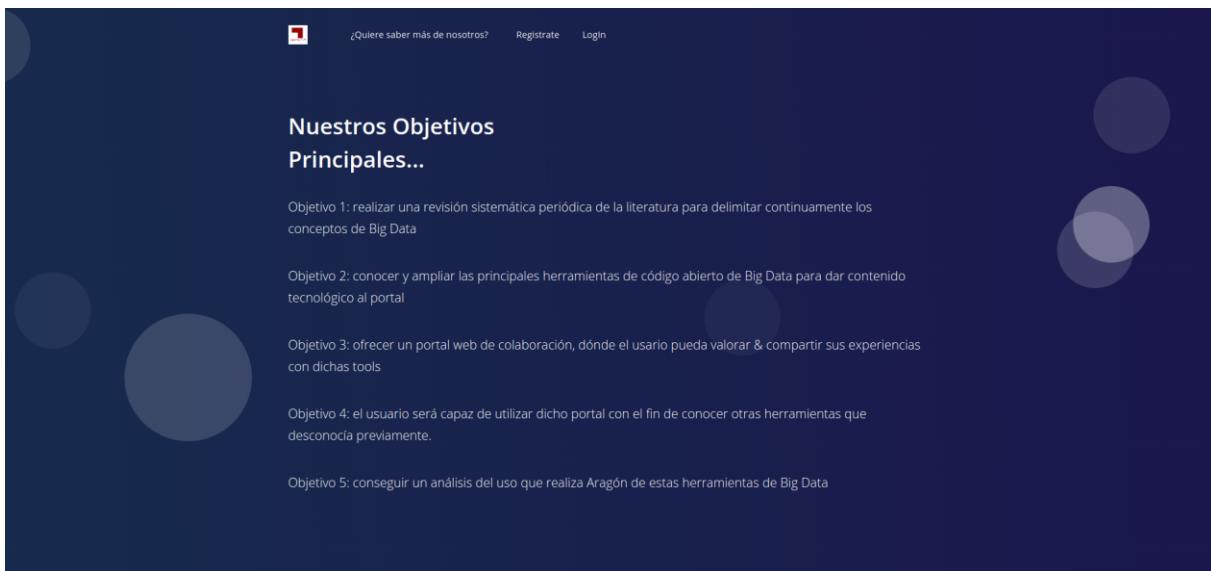


Figura 65 Pantalla: Objetivos.

Registro:

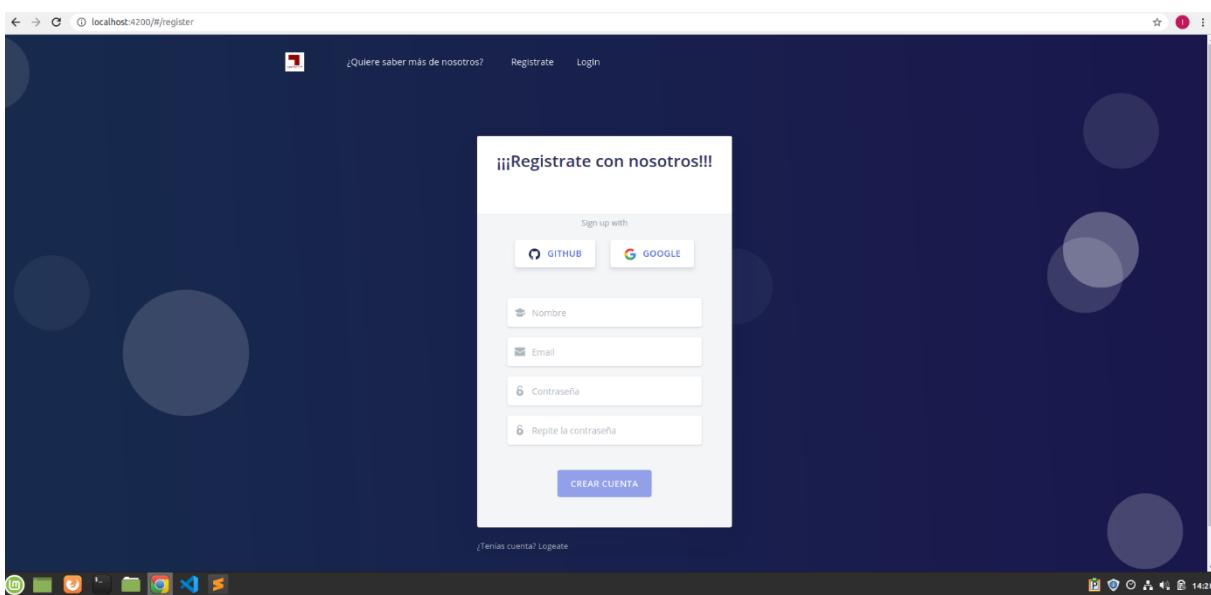


Figura 66 Pantalla: Registro.

Log In:

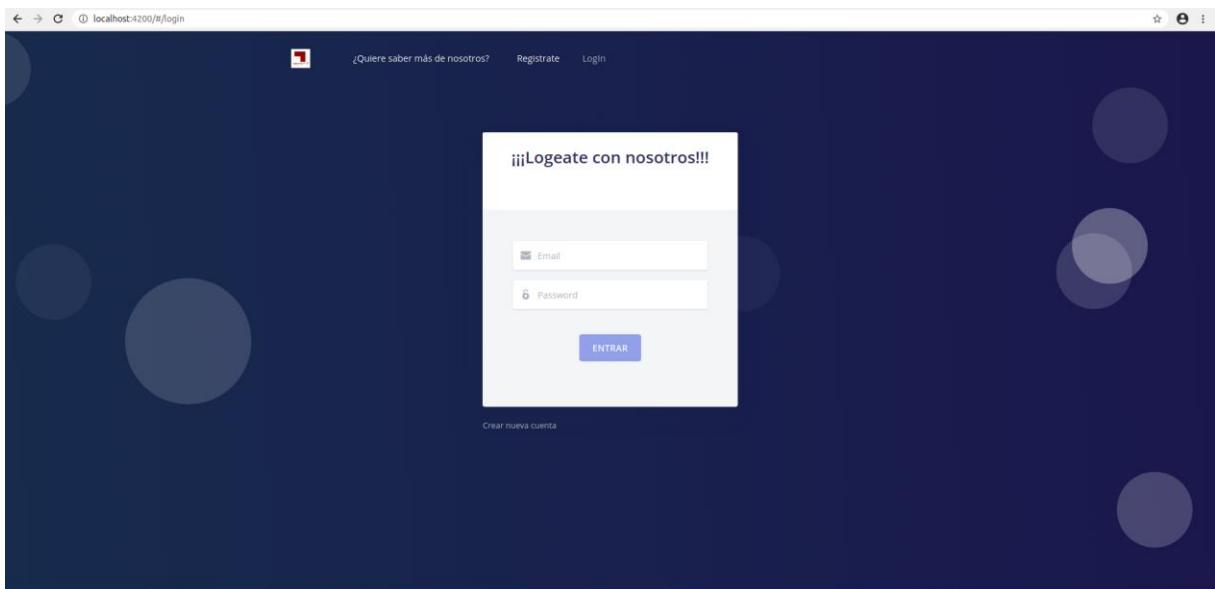


Figura 67 Pantalla: Inicio de Sesión.

Landing:

A screenshot of a web page titled "Local Tools...". The page has a sidebar on the left with a list of categories: Todas las categorías, Streaming, Framework, Query, Data access, Star Tools, Machine Learning, Búsqueda, Logging, Visualización, Colaboración, Seguridad, and Coordination. The main content area features three cards: "Streaming" (Categoría que recoge las aplicaciones de streaming), "Framework" (Categoría que recoge las aplicaciones de framework), and "Query" (Categoría que recoge las aplicaciones de query). Each card has a "Ver aplicaciones" button. To the right of the cards, there is an illustration of a person sitting at a desk with a computer monitor displaying a chart.

Figura 68 Pantalla: Categorías existentes

Application:

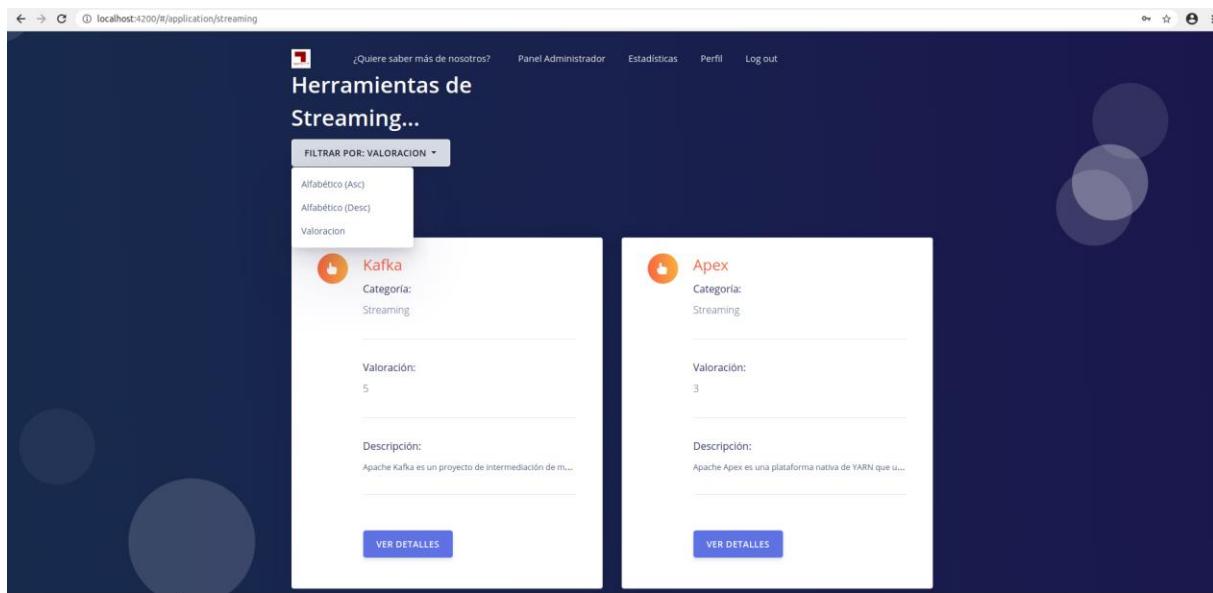


Figura 69 Pantalla: Aplicaciones de una categoría

Tool concreta:

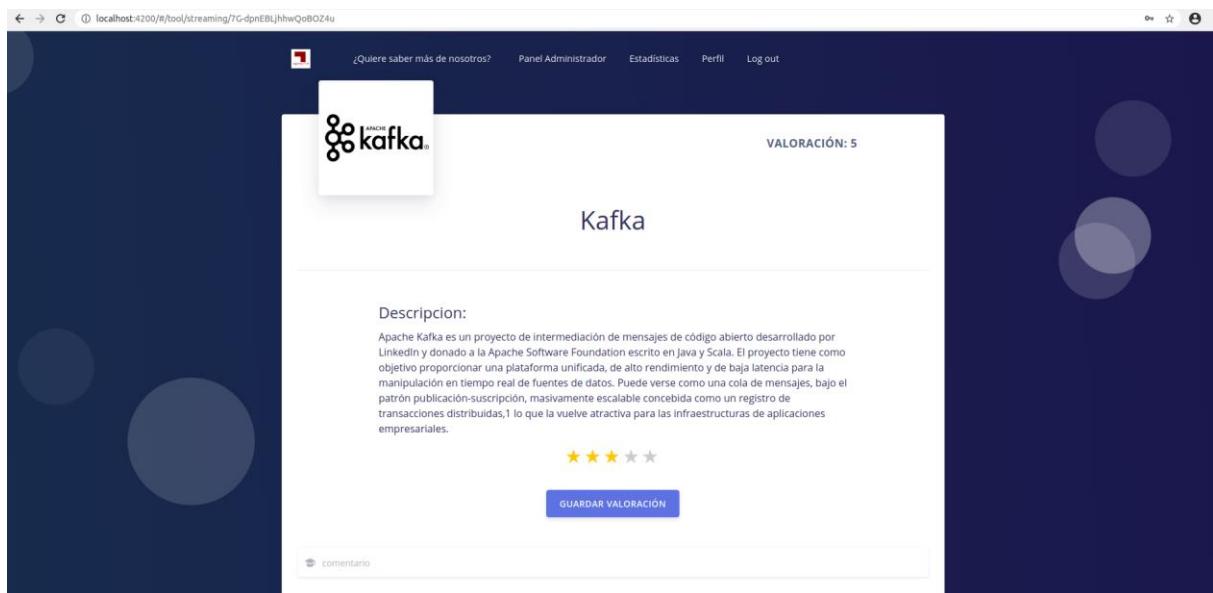


Figura 70 Pantalla: Herramienta concreta.

Estadísticas:



Figura 71 Pantalla: Estadísticas.

Perfil:

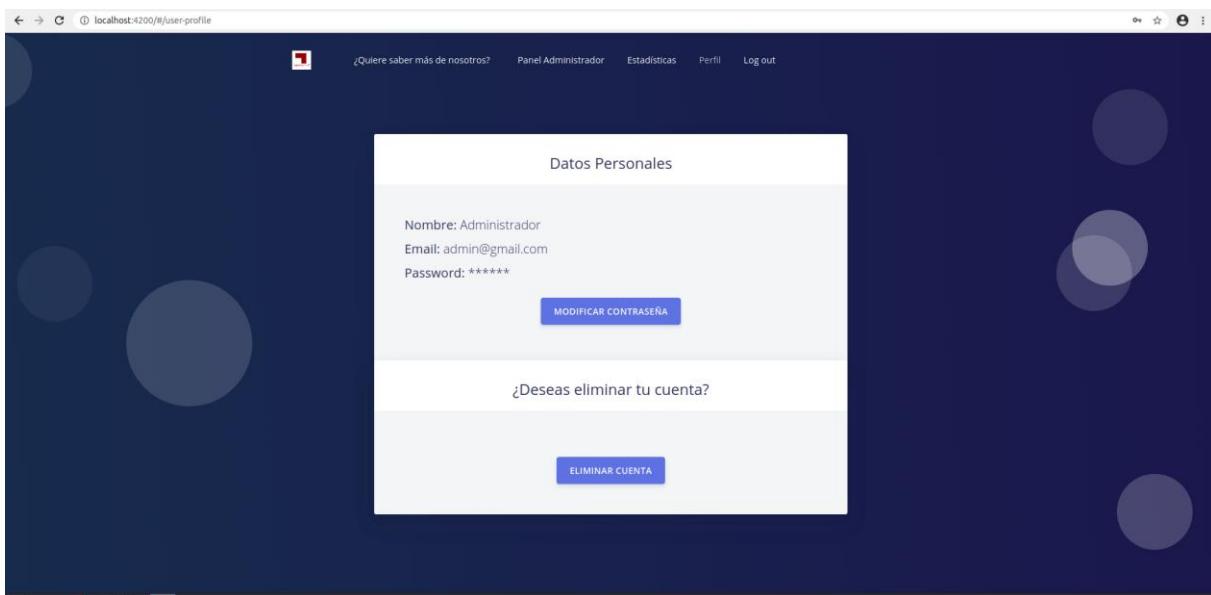


Figura 72 Pantalla: Perfil de Usuario.

Panel del Administrador:

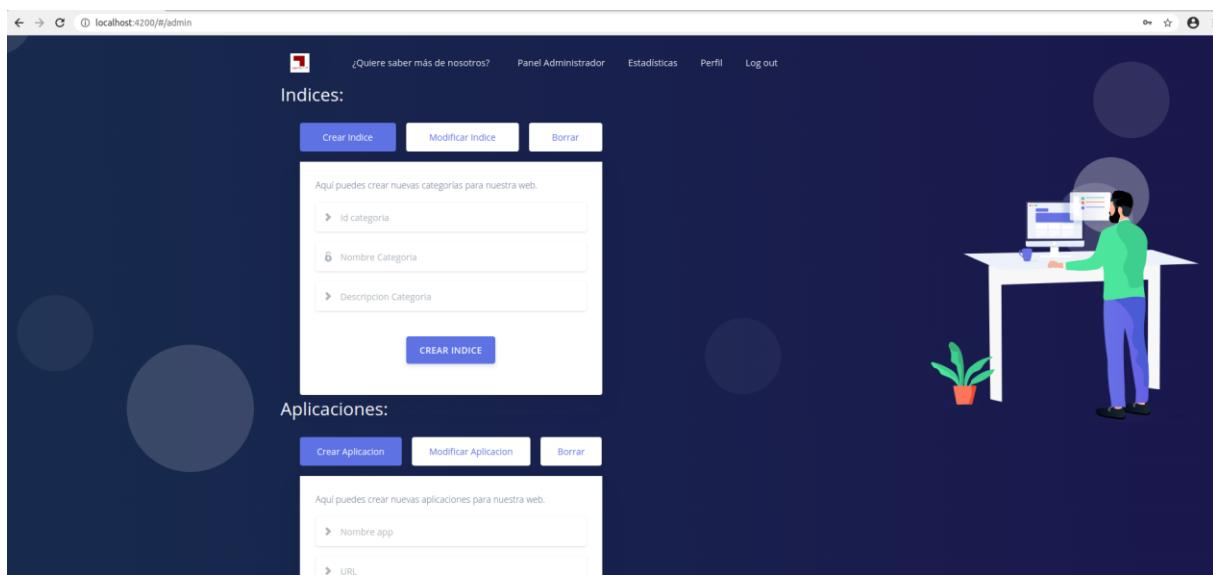


Figura 73 Pantalla: Panel de Administrador.

Creación de encuestas:

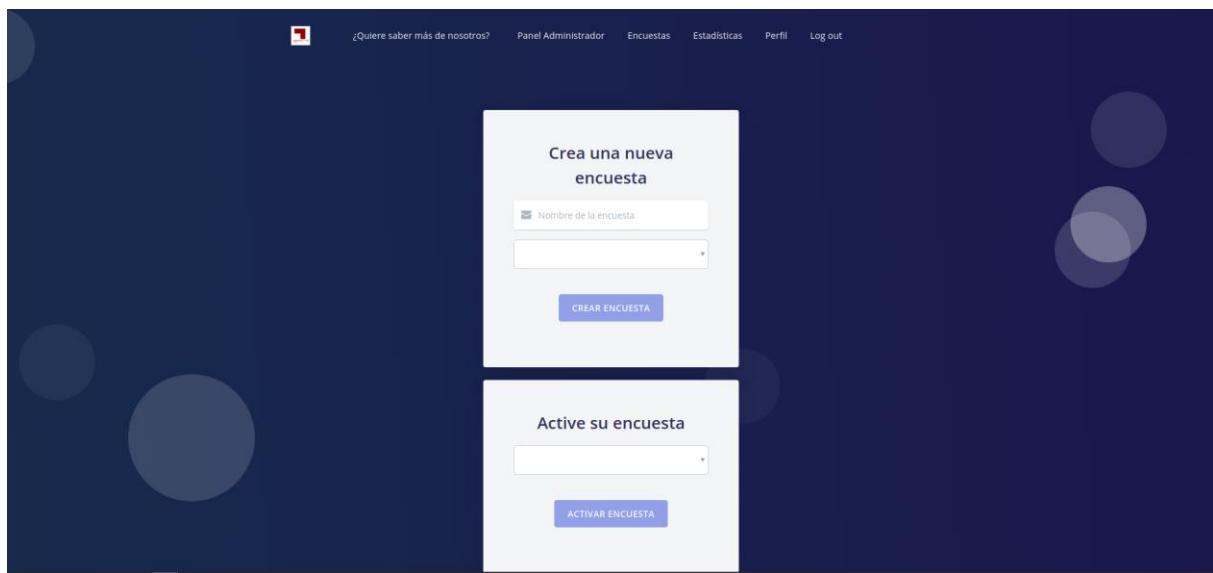


Figura 74 Pantalla: Creación/Activación encuestas

Creación de preguntas:

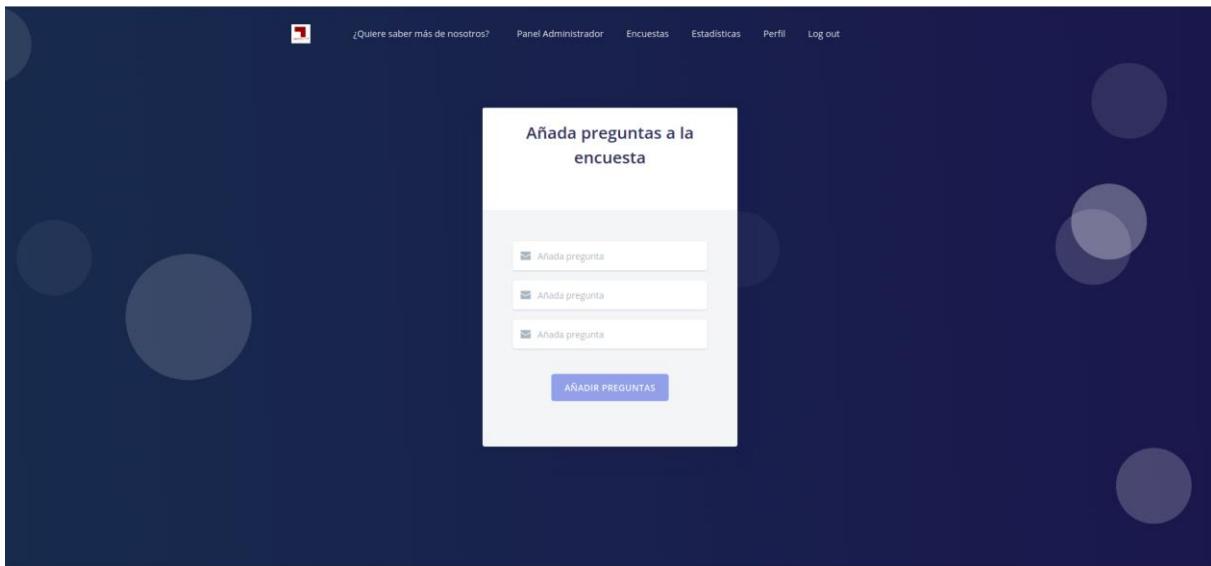


Figura 75 Pantalla: Creación de preguntas

Realización de encuesta activada:

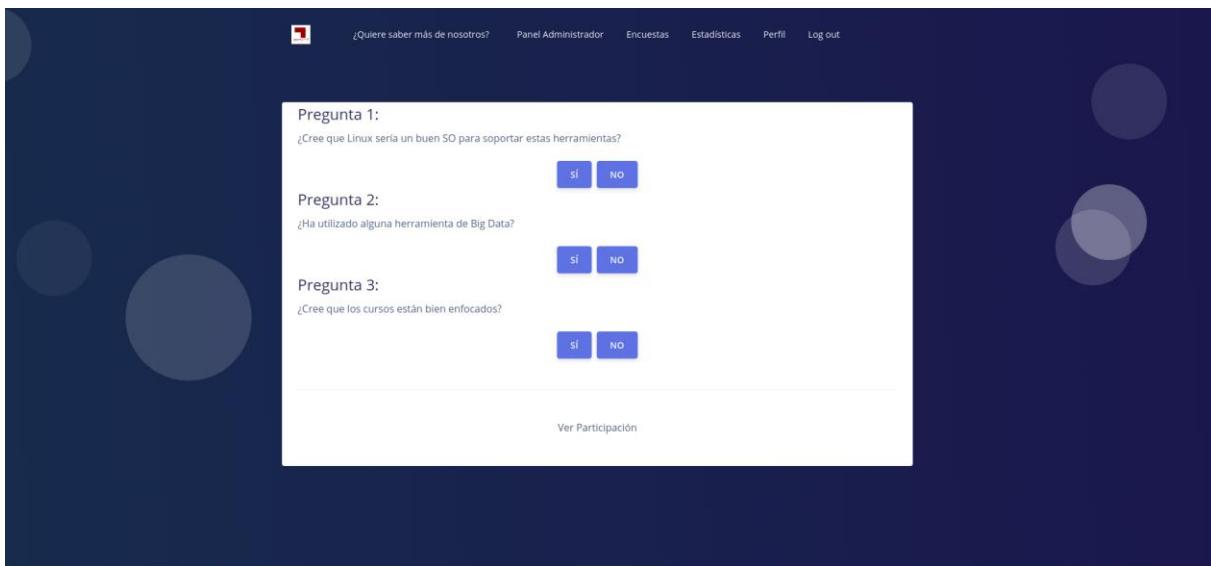


Figura 76 Pantalla: Responder Encuesta

Realización de encuesta desactivada:

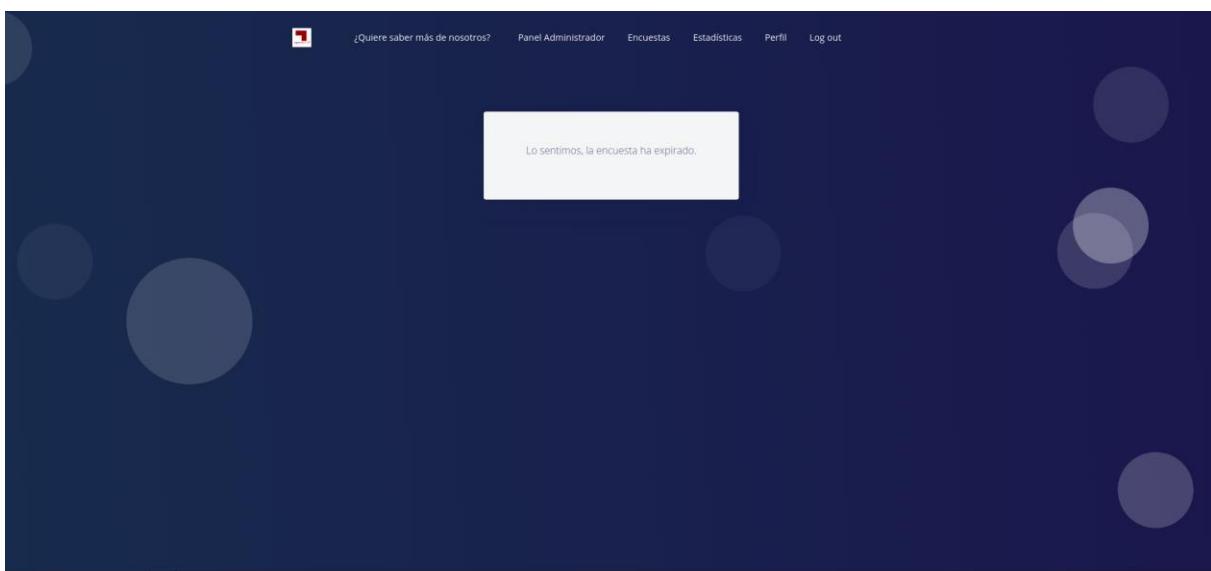


Figura 77 Pantalla: Encuesta Desactivada

M Guía de Despliegue

En esta sección se muestra la guía de despliegue, la cual resume los pasos a seguir, para desplegar adecuadamente el demostrador en los servidores de TECNARA. Este despliegue ha sido llevado a cabo por la empresa APSER, siguiendo los siguientes pasos:

INSTALAR NODE (a su vez se instalará el repositorio NPM): para ello se actualiza el repositorio y se instala el framework de Node.

```
sudo apt-get update
```

```
sudo apt-get upgrade
```

```
curl -sL https://deb.nodesource.com/setup_13.x | sudo bash
```

```
sudo apt-get install nodejs
```

INSTALAR DEPENDENCIAS DE AMBOS PROYECTOS: se debe descargar/clonar el proyecto de GitHub con sus dos directorios (web-tools-back & webTool_FrontEND). Se procederá a acceder a ambos directorios e instalar las dependencias que soportan, tales como librerías etc...

```
npm i
```

INSTALAR DOCKER POSTGRES: se crea el contenedor que soporta y contiene la Base De Datos.

```
sudo docker run --name webToolDB -p 5432:5432 -e POSTGRES_PASSWORD=<password> -d postgres:12.2
```

Acceder en el Backend al fichero de environment e introducir los datos configurados anteriormente, tales como la contraseña.

INSTALAR DOCKER ELASTICSEARCH: crear el contenedor con la imagen que posee el motor de búsqueda. Para ello:

```
sudo docker run -d --name elasticsearch -p 9200:9200 -p 9300:9300 -e "discovery.type=single-node" elasticsearch:7.6.2
```

INSTALAR DOCKER PORTAINER: crear un contenedor con la imagen de un gestor de contenedores para que sea más fácil la administración de estos (opcional)

```
sudo docker volume create portainer_data - sudo docker run -d -p 8000:8000 -p  
9000:9000 --name=portainer --restart=always -v  
/var/run/docker.sock:/var/run/docker.sock -v portainer_data:/data portainer/portainer
```

CARGA DE DATOS: en este punto tendríamos una Web Virgen. Es decir, sin categorías, ni usuarios, ni aplicaciones. Es por eso por lo que he creado la carpeta "CARGA DE DATOS" dentro del repositorio de GitHub. Que contiene un proyecto de Postman para ser importado y cargar directamente la BBDD. Tras dicho proceso la web tendrá las aplicaciones obtenidas tras la revisión sistemática.

Finalmente, se puede lanzar la web. Para ello se accedería a los directorios especificados a continuación y ejecutar los siguientes comandos:

[BackEnd]: *npm run start:dev*

[FrontEnd]: *sudo ng serve*